

平成24年度 修士論文

低歪み信号発生回路と
電源回路高効率化の研究

指導教員 小林 春夫 教授

群馬大学大学院工学研究科

電気電子工学専攻

高 川

目次

概要.....	4
第一章 低歪み信号発生回路	5
1.1 研究背景と目的	5
1.2 DAC 非線形性とその影響.....	6
1.2.1 理想的非線形性テスト	6
1.2.2 AWG 内部構造と非線形性特性	7
1.2.3 AWG 内部 DAC 非線形性の影響.....	8
1.2.4 内部 DAC の差動構成による偶数次歪の除去	9
1.2.5 研究の目標	10
1.3 従来任意信号発生器	11
1.3.1 従来信号発生器構造	11
1.3.2 単一正弦波発生 (AWG 内部 DAC に 3 次歪みがある場合)	11
1.3.3 2 トーン信号発生 (AWG 内部 DAC に 3 次歪みがある場合)	12
1.3.4 まとめ	14
1.4 提案信号発生手法 (位相差信号合成型)	15
1.4.1 位相差合成型構造と原理	15
1.4.2 単一正弦波発生 (AWG 内部 DAC に 3 次歪みがある場合)	16
1.4.3 2 トーン信号発生 (AWG 内部 DAC に 3 次歪みがある場合)	19
1.5 ダイナミックエレメントマッチング (DEM) 構成.....	23
1.6 位相差合成型+位相差切り替え型	26
1.6.1 「位相差合成型+位相差切り替え型」構造	26
1.6.2 「位相差合成型+位相差切り替え型」原理	28
1.6.3 「位相差合成型+位相差切り替え型」シミュレーション検証	29
1.7 まとめ.....	31
第二章 電源回路高効率化の研究	32
2.1 研究背景と目的.....	32
2.1.1 サーバ電源の高効率要求	32
2.1.2 電源回路の効率劣化原因	33
2.1.3 研究目標	34
2.1.4 研究環境	34
2.2 BLPFC AC/DC 回路 (Bridgeless Power Factor Correction AC/DC)	36
2.2.1 BLPFC AC/DC 回路開発ツール	36
2.2.2 BLPFC AC/DC 回路——リンク電圧の最適可変 (@ 50% Load).....	36
2.2.3 PFC AC/DC 回路——PWM(スイッチング周波数)の最適可変	38

2.3PSFB DC/DC 回路 (Phase Shift Full Bridge DC/DC Converter)	41
2.3.1 PSFB DC/DC 回路開発ツール.....	41
2.3.2PSFB DC/DC 回路——PWM(スイッチング周波数)最適可変(@10%~20% Load).....	41
2.4 まとめ.....	46
謝辞.....	47
発表論文.....	48
参考文献.....	49
付録:PSFB DC/DC 周波数可変プログラム.....	50

概要

本研究では、以下に示す二つのテーマを明らかにした。

① 低歪み信号発生回路

ミクストシグナル LSI テストの重要な項目としてアナログ回路部の線形性テストがある。これはアナログ回路の非線形性によって生成される歪成分を計測して線形性の要求を満たしているかのテストである。このテストには低歪のテスト信号が必要であり任意波形発生器 (AWG) にて生成する。低コストテスト化が重要であるので低価格で低パフォーマンス AWG (Arbitrary Waveform Generator: AWG) を使用することが多い。しかし、部品特性のばらつきが比較的大きいので非線形性が大きく、低歪の高精度テスト信号を生成することが難しい。そこで単一正弦波と 2 トーン信号を発生する AWG 内部 DAC のアーキテクチャについての検討を行なった。

② 電源回路高効率化の研究

スイッチング電源は直流安定化電源の一種で商用電源または直流電源を入力とし、これを半導体スイッチで高速にスイッチングして可聴周波数以上の高周波の交流に変換して再び整流平滑、安定化した直流電圧を得る電源である。小型・軽量で高効率を特徴とし情報機器や通信機器をはじめ、ほとんどの電子機器の電源として使用されている。サーバ電源は N+1 冗長運転方式で構成されている。したがって電源回路はフル負荷で作動することが稀で、普通 20%~50% 負荷で動作しているため軽負荷の電源効率向上が重視される。しかしながら、軽負荷の場合に高い効率を得るためには、回路トポロジーだけでは実現困難な非常に高いスペックが要求される。そこで我々は回路トポロジーを変えずに DSP を用いたデジタル制御を駆使した最適制御でハーフロード、或いは 20% 近くの軽負荷領域での効率の向上方法（プログラムの開発）を検討した。

第一章 低歪み信号発生回路

1.1 研究背景と目的

LSIを作るためには、極めて多くの工程が必要である。設計、試作、評価、量産など、これら全ての工程がさらに分化されている。それぞれのステップについて世界的な競争の中で新技術が開発されている。特に先端 LSI 開発ではテスト技術は上記の全ての工程に関係している。本来ならば設計と製造が完全であればテストは不要となる。しかしながら現代 LSI 設計の上でのプロセスパラメータの揺らぎは設計者に十分なマージンを与えてくれない。チップ内のトランジスタでさえも全て同じ動作する保障はない。そのため、あらゆる条件下で性能を確保する設計の実現は困難である。そこで LSI にテスト工程を投入することで製造段階での製造欠陥に起因する不具合を検出し、正しく動作するものが製造できる。特に先端の LSI 製品を生産する時、より不具合を検出する能力の高いテストを行うのが望まれる。しかしながら LSI テストコストの傾向としてはトランジスタあたりの製造コストは将来、大幅減少していくが、それに対してテストコストは増加すると思われる。低価格デバイスの大量生産するために、低コストのテストが必須であることが明白である。

テスト内容に重要な要素は線形性テストであり、あらゆる ADC、通信用デバイスの線形性テストは必須である。その線形性テストには歪みの少ないテスト信号が必要とされている。しかし、部品特性ばらつきによる非線形性が大きく、低歪みのテスト信号を生成することが困難である。

そこで本テーマでは低コスト、かつ高性能の線形性テストの実現を目的とした。

1.2 DAC 非線形性とその影響

1.2.1 理想的非線形性テスト

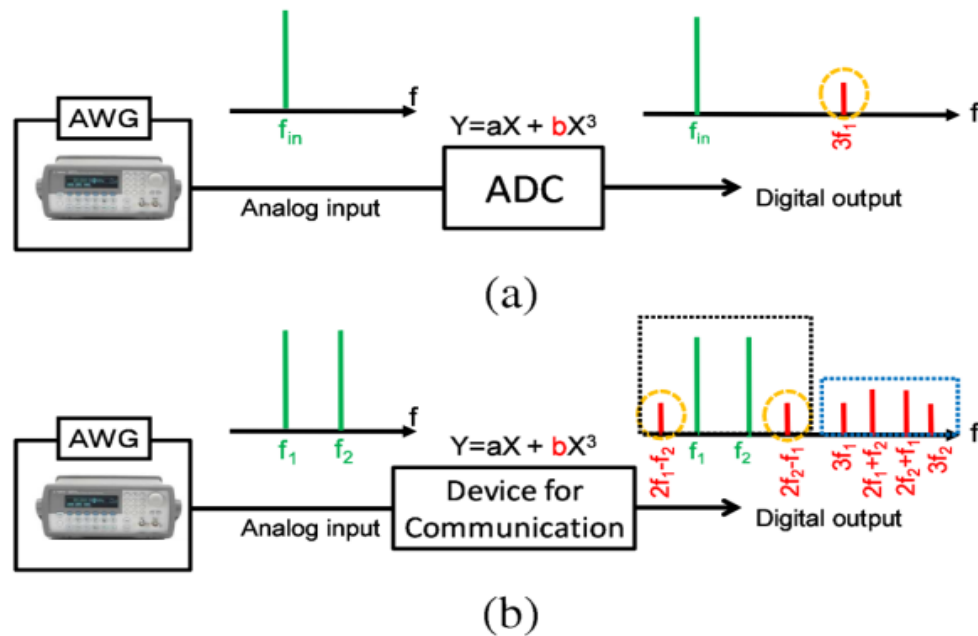


図 1.2.1 AWG によるテスト信号生成と ADC と通信デバイス非線形性テスト.
(a) 単一正弦波の場合. (b) 2 トーン信号の場合.

図 1.2.1 が示すように、理想的な ADC や通信デバイスの線形性テストの流れは「綺麗な」すなわち歪みが小さいテスト信号を AWG から発生させ、ADC や通信デバイスに入力し、そのデジタル出力成分の歪みを計る事である。その歪み成分の分析結果に基づき ADC や通信デバイスの線形性性能を評価する。

(AWG: Arbitrary Waveform Generator (任意波形発生器))

1.2.2 AWG 内部構造と非線形性特性

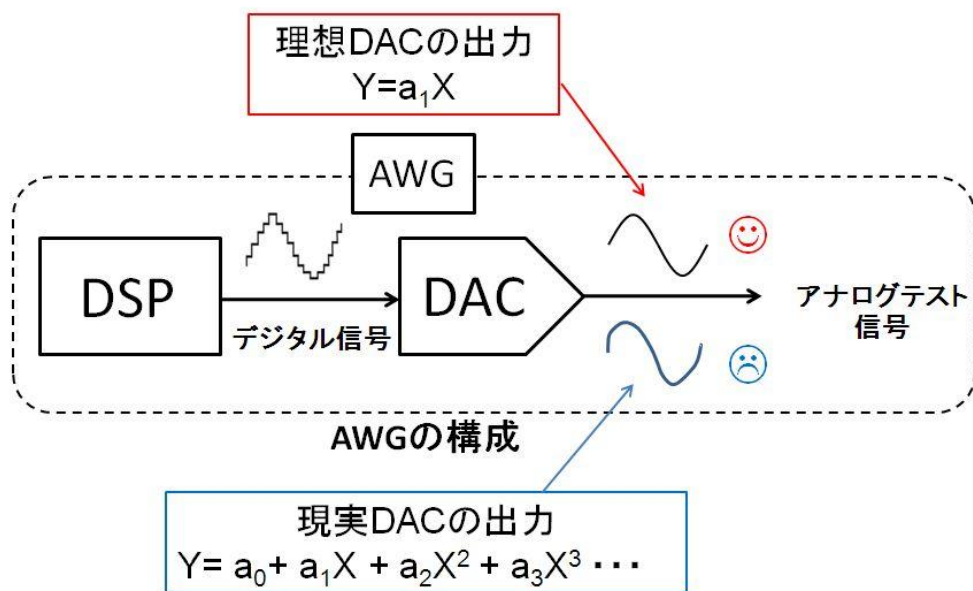
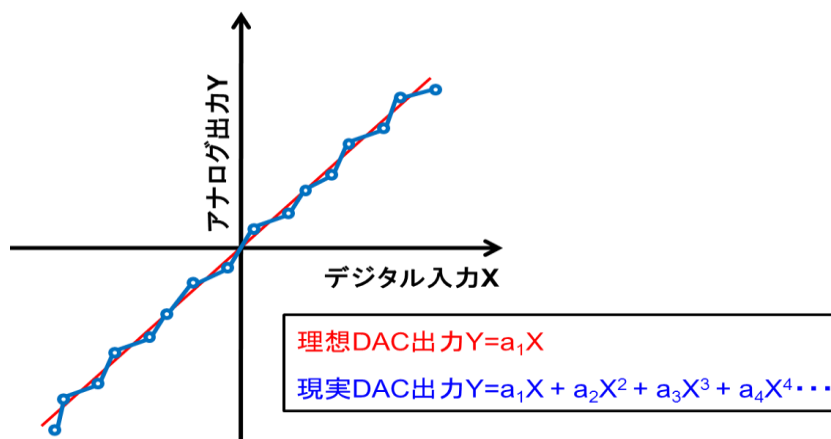


図 1.2.2 AWG(任意信号発生器)内部構造

図 1.2.2 示すように点線に囲まれた部分は AWG の内部構造となる。DSP からデジタル信号が生成され、DAC に入力する。DAC の出力はアナログテスト信号となる。理想の DAC 出力特性は線形であるため、出力信号は歪みのないアナログ信号を出すことができる。しかしながら現実の DAC 特性は素子のばらつきにより完全の線形ではなく二次や三次歪み成分を含む。そのため、DAC 出力のアナログテスト信号は歪んでいる。



1.2.3 DAC の非線形性

1.2.3 AWG 内部 DAC 非線形性の影響

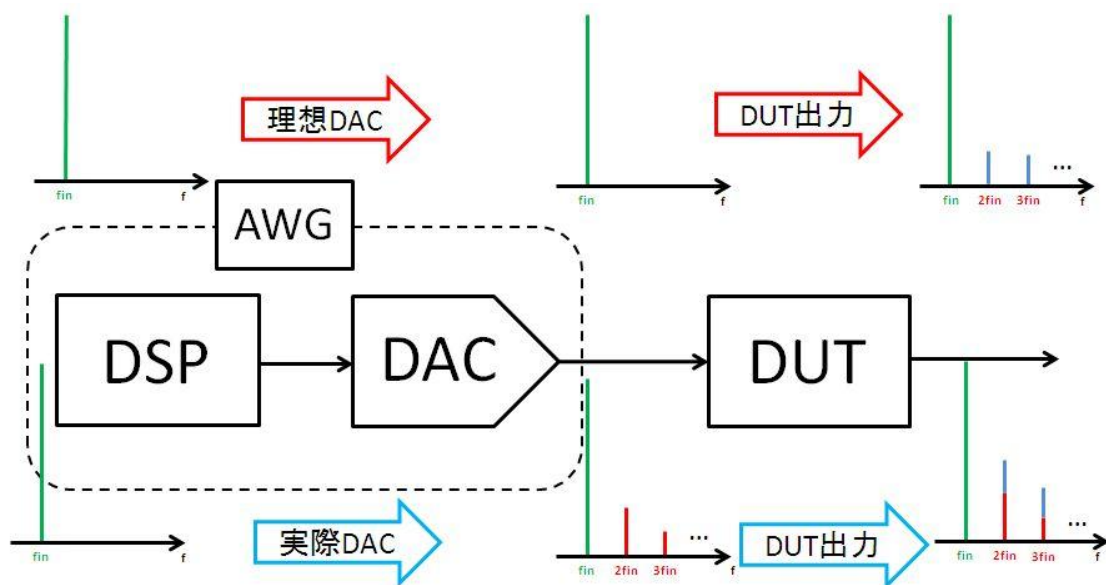


図 1.2.4 AWG 内部 DAC 線形と非線形の場合(DUT :Device Under Test)

図 1.2.4 が示すように AWG 内部 DAC が理想の場合、DUT の出力 FFT 分析による歪み成分の測定は正確である。しかし実際に AWG 内部 DAC に非線形性特性があるため DUT の出力 FFT 分析に歪み成分で測定結果に誤差が出る。従って AWG から出るアナログテスト信号の歪み成分の多少は AWG の性能を大きく左右する。

1.2.4 内部 DAC の差動構成による偶数次歪の除去

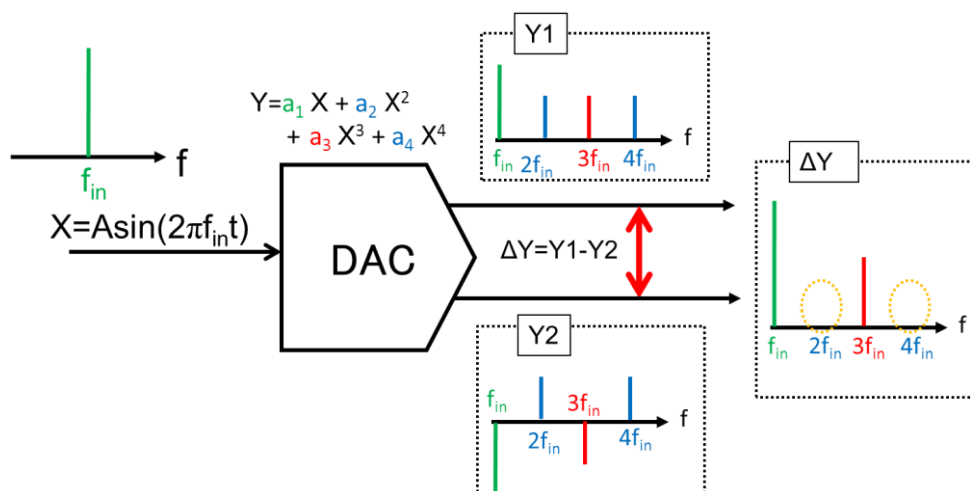


図 1.2.5 偶数次歪を除去するための内部 DAC の差動構成

図 1.2.5 が示したように DAC 出力を差動構成にすることで偶数次によって生成される歪み成分をキャンセルすることが可能である。 Y_1 が従来の出力であり Y_2 が 180° 位相変化した出力である。 Y_2 では基本波 f_{in} が 180° 位相変化、HD2 $2f_{in}$ が 360° の位相変化、HD3 $3f_{in}$ が 540° の位相変化、HD4 $4f_{in}$ が 720° の位相変化をする。このように次数が大きくなるごとに 180° 位相変化をする。つまり基本波と全ての奇数次の HD は 180° の位相変化をし全ての偶数次の HD は 360° の位相変化なので影響は無い。最終的な出力は $\Delta Y = Y_1 - Y_2$ なので基本波と奇数次の HD は 2 倍となり偶数次の HD はキャンセルされる。この場合、単一正弦波入力時でも HD2 がキャンセルされるため HD3 が最も問題となる。そのため単一正弦波発生の場合でも DAC に 2 次歪がある場合ではなく DAC に 3 次歪がある場合を優先的に検討する。

1.2.5 研究の目標

本研究の目的は「低コスト且つ高精度な線形性テストの実現」である。この実現のために今回は、AWG 内部の DAC アーキテクチャを変えることによって低コスト・低パフォーマンス AWG を使用しても高精度で低歪みのアナログテスト信号を発生させることが可能な研究である。

低歪み成分の信号を生成するためには AWG の出力の歪み成分をキャンセルする必要がある。そのために位相差信号合成型という信号発生手法が提案されている。位相差信号合成型は AWG の新アーキテクチャ方式で、歪み成分をキャンセルする手法である。

下記の従来信号発生手法と提案した信号発生手法の検討を行う。検討にはそれぞれ二つの場合について行っていく。

- ① 単一正弦波発生 (AWG 内部 DAC に 3 次歪みがある場合)
- ② 2 トーン信号発生 (AWG 内部 DAC に 3 次歪みがある場合)

提案手法を、さらに二つの場合に分けて検討をした。

- ① 単一正弦波発生 (AWG 内部 DAC に 3 次歪みがある場合)
 - DAC 特性間にばらつきがない場合
 - DAC 特性間にばらつきがある場合
- ② 2 トーン信号発生 (AWG 内部 DAC に 3 次歪みがある場合)
 - DAC 特性間にばらつきがない場合
 - DAC 特性間にばらつきがある場合

1.3 従来任意信号発生器

1.3.1 従来信号発生器構造

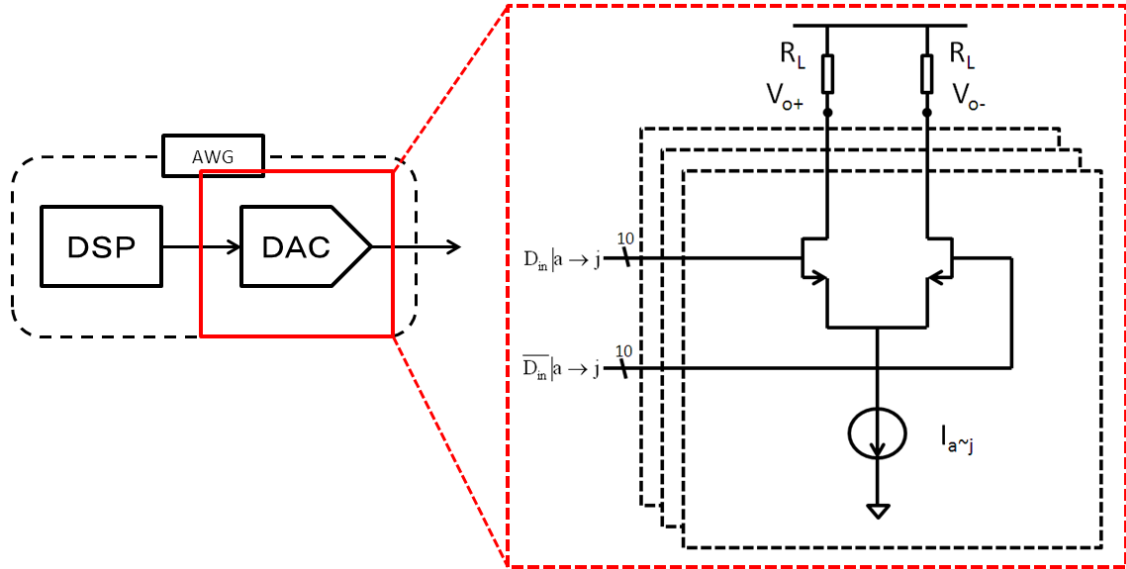


図 1.3.1 従来任意信号発生器内部 DAC の構造

図 1.3.1 が示したのが 10-bit 作動二進重み付けの DAC を用いた従来任意発生器の内部構成である。DAC の出力は $V_{out}=V_{o+}-V_{o-}$ 、電流セルの電流源は $I_{a\sim j}$ 、それぞれ 1,2,4,8,16,32,64,128,256,512 になる。

1.3.2 単一正弦波発生（AWG 内部 DAC に 3 次歪みがある場合）

DAC に 3 次歪みがある AWG を使用して単一正弦波を発生させる場合を考える。ここでは簡単にするため DAC 出力特性を式 1.3.1 に示すように近似する。DAC のサンプリング周期を T_s 、そのサンプリング周波数を f_s とする ($T_s f_s = 1$)。サンプリング周期 T_s とサンプリング周波数 f_s は以後共通とする。入力信号 X として式 1.3.2 のような単一正弦波を入力する。式 1.3.2 を式 1.3.1 へ代入すると、出力信号 Y として式 1.3.3 のようになる。

$$\text{DAC 出力特性} \quad Y = aX + bX^3 \quad 1.3.1$$

$$\text{入力信号} \quad X = A \sin 2\pi f_{in} n T_s \quad 1.3.2$$

$$\text{出力信号} \quad Y = \left(aA + \frac{3bA^3}{4}\right) \sin 2\pi f_{in} n T_s - \frac{bA^3}{4} \sin 2\pi 3f_{in} n T_s \quad 1.3.3$$

出力信号 Y(式 1.3.3)には入力信号 X(式 1.3.1)には存在しない歪成分 $HD3(3f_{in})$ が生成される。これをブロック図で示したものは図 1.3.2 である。

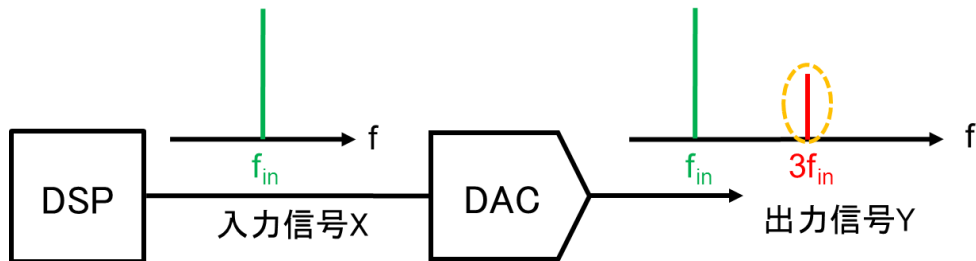


図 1.3.2 単一正弦波発生(DAC に 3 次歪がある場合)

この出力信号 Y の中にある歪み成分を SPICE シミュレーションより検証した。

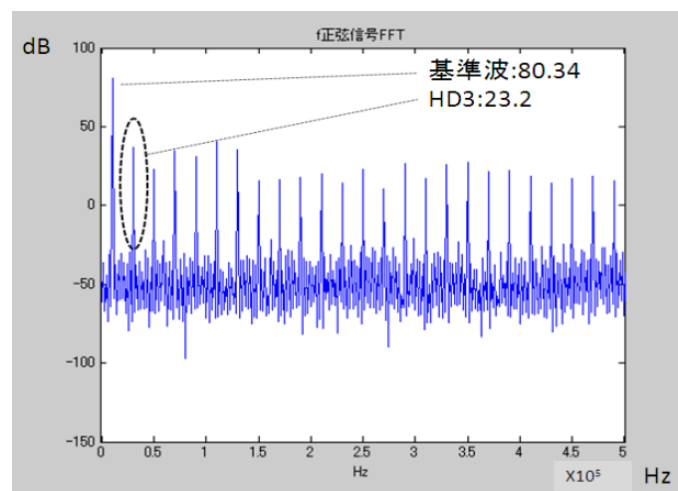


図 1.3.3 従来手法（単一正弦波発生）

図 1.3.3 が示したように、DAC の非線形特性により、基本波近傍に HD3 が生成する。

1.3.3 2 トーン信号発生（AWG 内部 DAC に 3 次歪みがある場合）

DAC に 3 次歪がある AWG を使用して、2 トーン信号を発生させる場合を考える。ここでは簡単にするため DAC 出力特性を式 1.3.4 に示すように近似する。入力信号 X として式 1.3.5 のような 2 トーン信号を入力する。式 1.3.5 を式 1.3.4 へ代入すると出力信号 Y として、式 1.3.6 のようになる。

$$\text{DAC 出力特性 } Y = aX + bX^3 \quad 1.3.4$$

$$\text{入力信号 } X = A \sin 2\pi f_1 T_s + B \sin 2\pi f_2 T_s \quad 1.3.5$$

$$\begin{aligned} \text{出力信号 } Y = & (aA + \frac{3}{4}bA^3 + \frac{3}{4}bAB^2) \sin 2\pi f_1 nT_s \\ & + (aB + \frac{3}{4}bB^3 + \frac{3}{4}bA^2B) \sin 2\pi f_2 nT_s \\ & - \frac{3}{4}bA^2B \sin 2\pi(2f_1 + f_2)nT_s + \frac{3}{4}bA^2B \sin 2\pi(2f_1 - f_2)nT_s \\ & - \frac{3}{4}bAB^2 \sin 2\pi(2f_2 + f_1)nT_s + \frac{3}{4}bAB^2 \sin 2\pi(2f_2 - f_1)nT_s \\ & - \frac{1}{4}bA^3 \sin 2\pi 3f_1 nT_s - \frac{1}{4}bB^3 \sin 2\pi 3f_2 nT_s \end{aligned} \quad 1.3.6$$

出力信号 Y(式 1.3.6)には入力信号 X(式 1.3.4)には存在しない歪成分 HD3(3f₁、3f₂)と IMD3(2f₁+f₂、2f₂+f₁、2f₁-f₂、2f₂-f₁)が生成される。この中でも 2f₁-f₂と 2f₂-f₁の IMD3 が入力信号近傍に発生するため問題となる。これをブロック図で示したものは図 1.3.3 である。

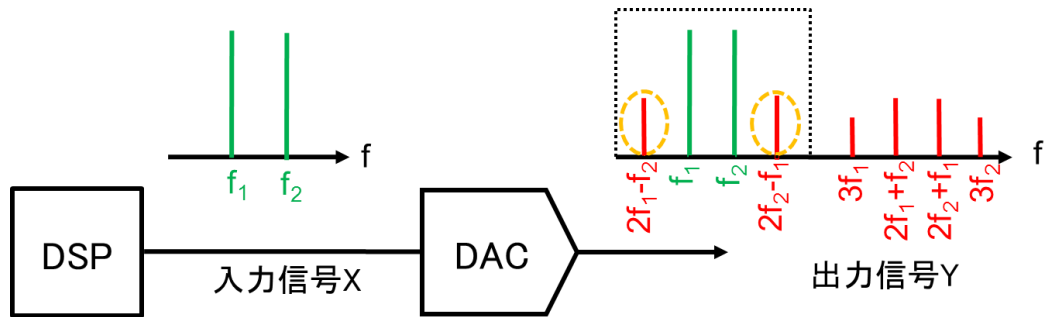
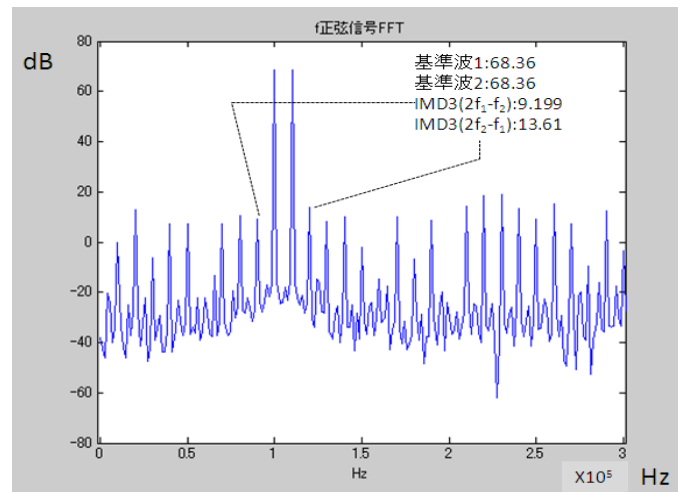


図 1.3.4 単一正弦波発生(DAC に 3 次歪がある場合)

この出力信号 Y の中にある歪み成分を SPICE シミュレーションより検証した。



従来手法（2 トーン信号発生）

図 1.3.3 が示したように、DAC の非線形性特性により、基本波近傍に HD3 が生成される。

1.3.4 まとめ

従来手法の任意信号発生器内部 DAC の非線形性特性（3 次歪み）より

単一正弦波を生成する場合、DSP から単一正弦波を DAC へ入力すると出力に歪み成分 HD3 が生成される。

2 トーン信号を生成する場合、DSP から 2 トーン信号を DAC へ入力すると出力に歪み成分 HD3 と IMD3 が発生する。また IMD3 が入力周波数近傍に発生するので基本信号に与える影響が HD3 より大きい。

1.4 提案信号発生手法（位相差信号合成型）

1.4.1 位相差合成型構造と原理

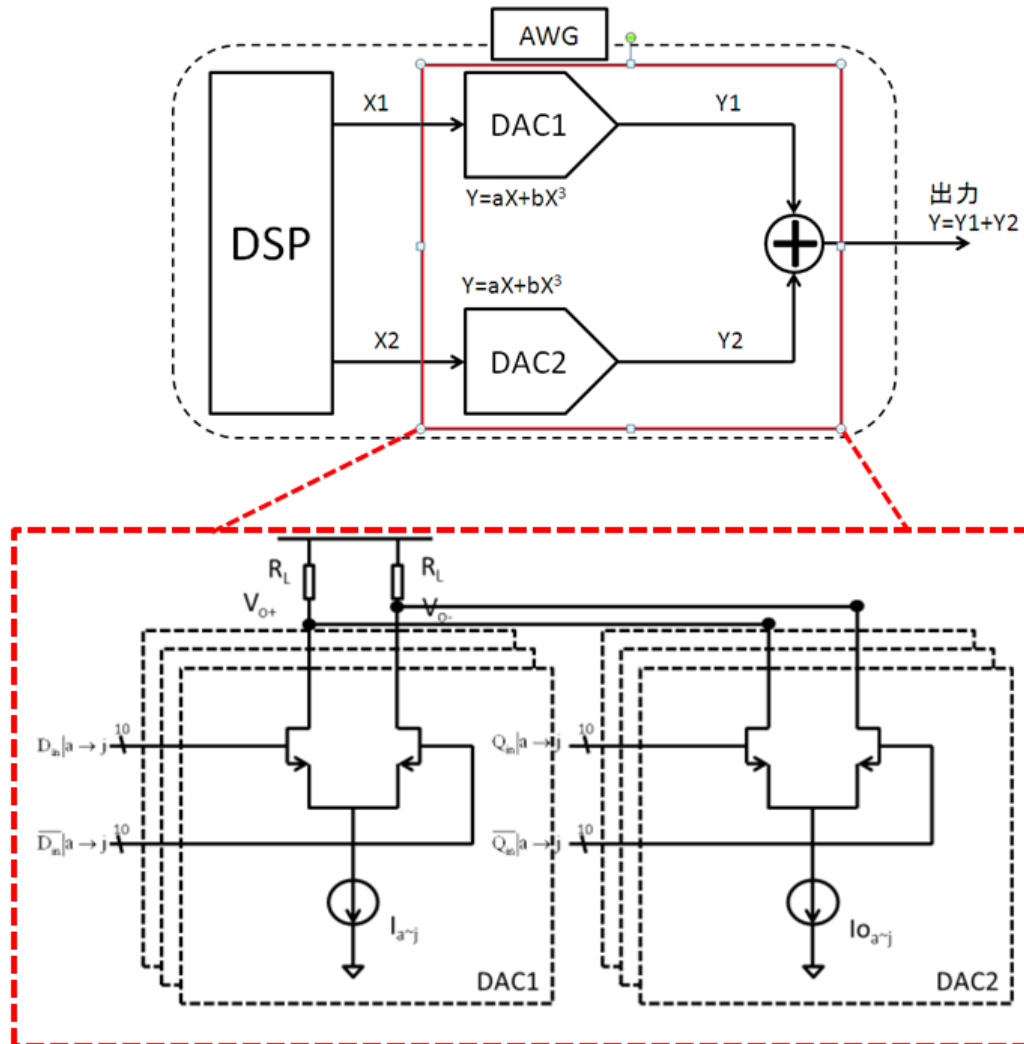


図 1.4.1 位相差合成型を用いた AWG 内部の DAC 構成

図 1.4.1 が示したように、一つの 10-bit 作動二進重み付けの DAC を用いた従来任意発生器の内部構成です。DAC の出力は $V_{out}=V_{o+}-V_{o-}$ 。電流セルの電流源は I_{a-j} と I_{o-a-j} (10 個), それぞれ 1,2I,4I,8I,16I,32I,64I,128I,256I,512I になる。AWG の内部 DAC が非線形性であると単一正弦波入力, 2 トーン信号入力の両方の場合において AWG 出力に歪み成分が生成される。その改善のために DAC を複数使用し出力を加算する構成を提案した。一方の DAC にデジタル入力 X_1 , もう一方の DAC に X_2 を入力する。2 つの DAC からそれぞれ位相差を持ったアナログ出力 Y_1 と Y_2 が出力され加算器によって Y_1 と Y_2 を加算し最終的に歪成分がキャンセルされた出力 Y を得る。キャンセルできるのはデジタル入力 X_1 と X_2 に位相差を持たせるためである。出力 Y の基本波

は DAC の 1 次係数の項によって生成されるが歪成分は DAC の 2 次係数や 3 次係数等の項によって生成される。このため基本波は X_1 と X_2 の位相差分となり、歪成分は基本波の 2 倍や 3 倍の位相差となる。この特徴を利用して歪成分のみキャンセル可能な位相差をデジタル入力 X_1 と X_2 に与えている。

1.4.2 単一正弦波発生（AWG 内部 DAC に 3 次歪みがある場合）

従来手法で DAC に 3 次歪がある AWG で単一正弦波を発生させると歪成分 HD3 が生成されることが上記で確認されている。そこで HD3 をキャンセルするために DAC2 つ使用した位相差信号合成型を適用する。ここでは簡単のため DAC1、2 の出力特性を式 1.4.1 に示すように近似する。入力信号 X_1 として式 1.4.2 のような $\pi/6$ 位相の進んだ単一正弦波、入力信号 X_2 として式 1.4.3 のような $\pi/6$ 位相の遅れた単一正弦波を式 1.4.1 へ代入し加算すると出力信号 Y として式 1.4.4 のようになる。

$$\text{DAC1,2 出力特性 } Y_{1,2} = aX + bX^3 \quad 1.4.1$$

$$\text{入力信号 } X_1 = A \sin(2\pi f_{in} nT_s + \pi/6) \quad 1.4.2$$

$$\text{入力信号 } X_2 = A \sin(2\pi f_{in} nT_s - \pi/6) \quad 1.4.3$$

$$\text{出力信号 } Y = Y_1 + Y_2 = 1.7(aA + \frac{3bA^3}{4}) \sin 2\pi f_{in} nT_s \quad 1.4.4$$

位相差信号合成型の出力信号 Y (式 1.4.4) を見ると従来手法の出力信号 Y (式 1.3.3) で存在した HD3($3f_{in}$) がキャンセルされていることがわかる。これをブロック図で示したのが図 1.4.2 である。

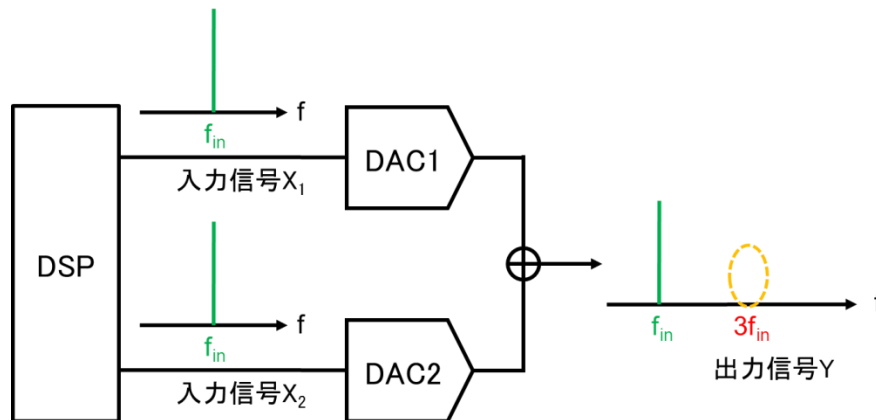


図 1.4.2 単一正弦波発生(3 次歪成分打消し)

DAC 出力信号 Y_1 と Y_2 の基本波は位相差が $\pi/3$ となる。よって図 1.4.3(a)のように出力信号 Y はベクトル合成される。一方、DAC 出力信号 Y_1 と Y_2 の歪成分 HD3 は位相差が基本波の 3 倍の π となる。よって図 1.4.3(b)のようにキャンセルされる。

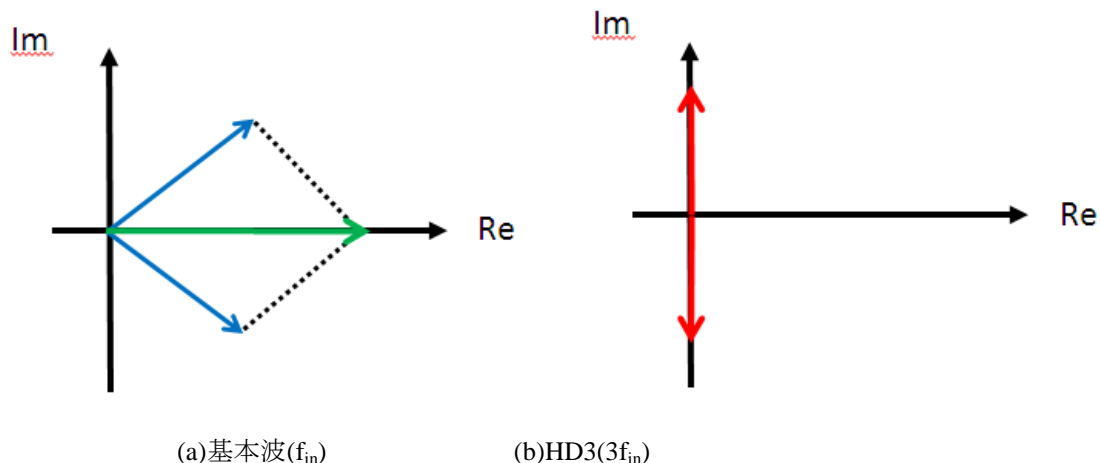


図 1.4.3 HD3 のキャンセル効果

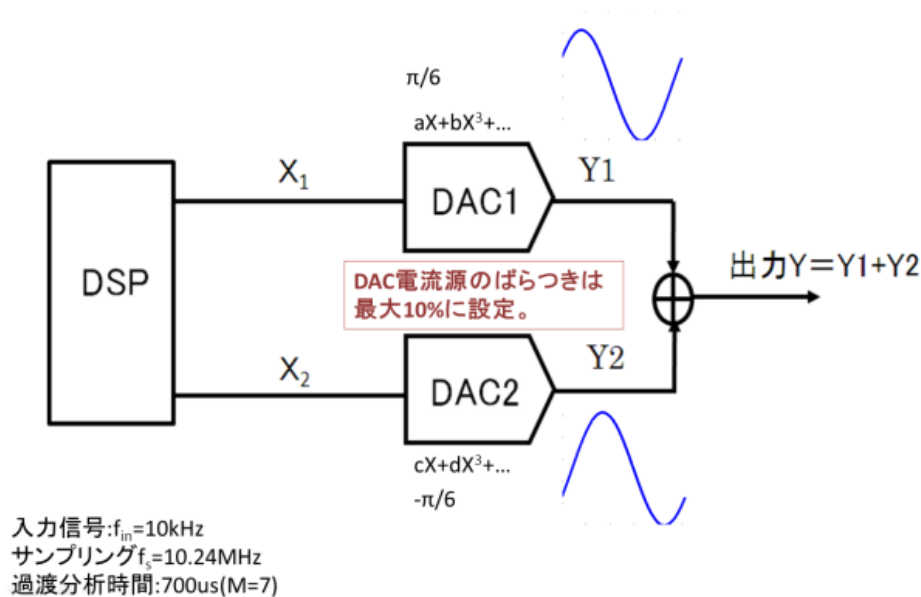
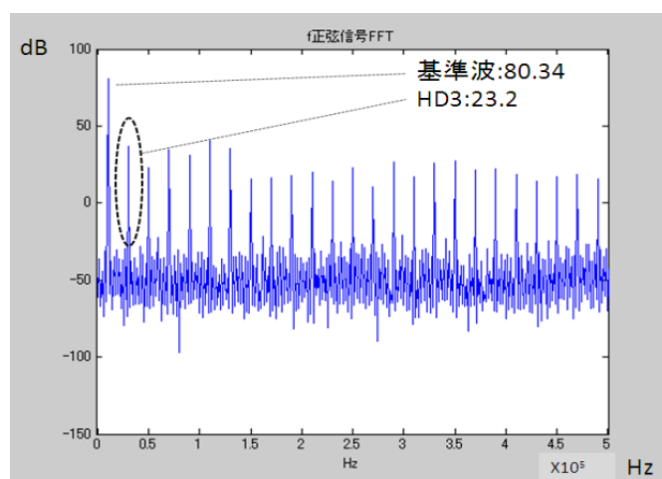


図 1.4.4 位相差合成型・単一正弦波 HD3 キャンセル

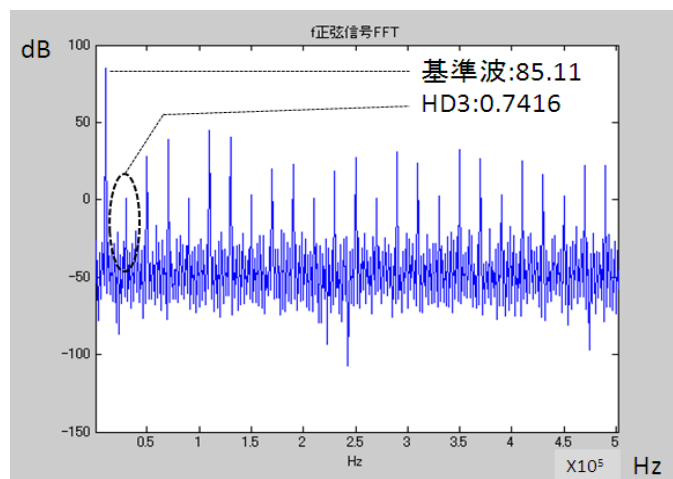
図 1.4.4 の条件に基づき、3 次歪みキャンセル (HD3) 効果を SPICE のシミュレーションで確認した。

DAC 特性間にばらつきがない場合: $b = d$;

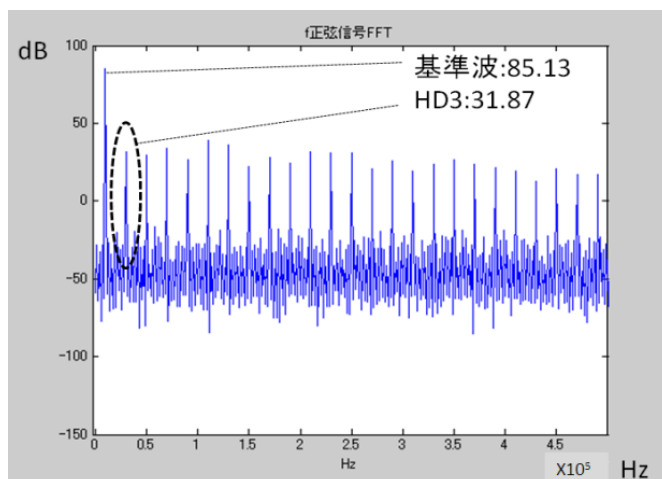
DAC 特性間にばらつきがある場合: $b \neq d$;



(a) 従来手法



(b) 提案手法：DAC 特性間にばらつきがない場合



(c) 提案手法：DAC 特性間にばらつきがある場合

図 1.4.5 提案構成によるシミュレーション結果・単一正弦波

図 1.4.5 が示したように、二つの DAC 中の電流セルの電流源のミスマッチ量が同じなら、(b)のように従来手法(a)より HD3 を減少できる。二つの DAC 中の電流セルの電流源のミスマッチ量が違うなら(c)のように HD3 はキャンセルできない。

1.4.3 2 トーン信号発生（AWG 内部 DAC に 3 次歪みがある場合）

従来手法で DAC に 3 次歪みがある AWG で 2 トーン信号を発生させると信号帯域近傍に歪成分 IMD3 が生成されることが上記で確認されている。そこで IMD3 をキャンセルするために DAC を 2 つ使用した位相差信号合成型を適用する。ここでは簡単化して DAC1、2 の出力特性を式 1.4.5 に示すように近似した。入力信号 X1 として式 1.4.6 のように 2 つのうち片方の正弦波を $\pi/6$ 位相を進ませ、もう片方の正弦波を $\pi/6$ 遅らせた信号を入力する。入力信号 X2 として式 1.4.7 のような入力信号 X1 の逆の位相差を持つ信号を入力する。次に式 1.4.6 と式 1.4.7 を式 1.4.5 に代入し加算すると出力信号 Y として式 1.4.8 になる。

$$\text{DAC1,2 出力特性 } Y_{1,2} = aX + bX^3 \quad 1.4.5$$

$$\text{入力信号 } X_1 = A \sin(2\pi f_1 nT_s + \pi/6) + B \sin(2\pi f_2 nT_s - \pi/6) \quad 1.4.6$$

$$\text{入力信号 } X_2 = A \sin(2\pi f_1 nT_s - \pi/6) + B \sin(2\pi f_2 nT_s + \pi/6) \quad 1.4.7$$

$$\begin{aligned} \text{出力信号 } Y = Y_1 + Y_2 = & 1.7(aA + \frac{3bA^3}{4} + \frac{3}{2}bAB^2) \sin 2\pi f_1 nT_s \\ & + 1.7(aB + \frac{3bB^3}{4} + \frac{3}{2}bA^2B) \sin 2\pi f_2 nT_s \\ & - 1.3bA^2B \sin 2\pi(2f_1 + f_2)nT_s \\ & - 1.3bAB^2 \sin 2\pi(2f_2 + f_1)nT_s \end{aligned} \quad 1.4.8$$

位相差信号合成型の出力信号 Y(式 1.4.8)を見ると従来手法の出力信号 Y(式 1.3.6)で存在した HD3($3f_1 + 3f_2$)と IMD3($2f_1 - f_2 + 2f_2 - f_1$)がキャンセルされていることがわかる。これをブロック図で示したものが図 3.3.1 である。

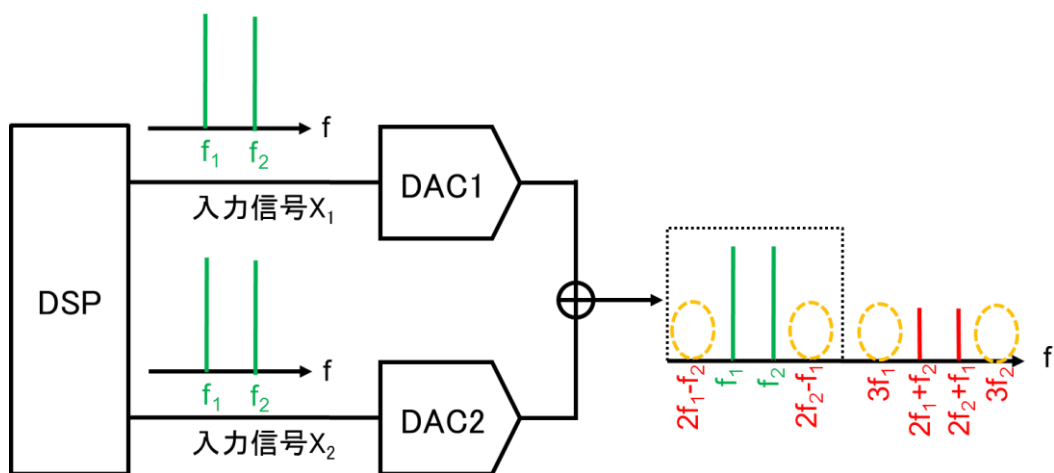


図 1.4.6 2 トーン信号発生(3 次歪成分打消し)

DAC 出力信号 Y_1 と Y_2 の歪成分 HD3 と IMD3 は位相差が基本波の 3 倍の π となる。よって図 1.4.7 のようにキャンセルされる。

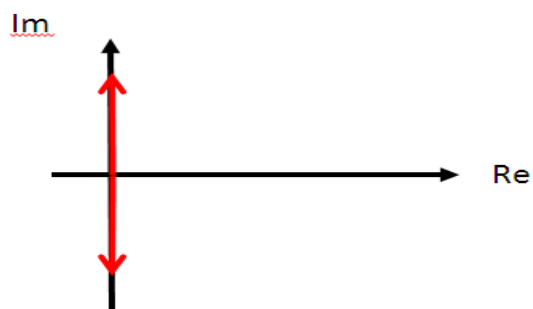


図 1.4.7 HD3・IM3 のキャンセル効果{ HD3($3f_1, 3f_2$)、IM3($2f_1-f_2, 2f_2-f_1$) }

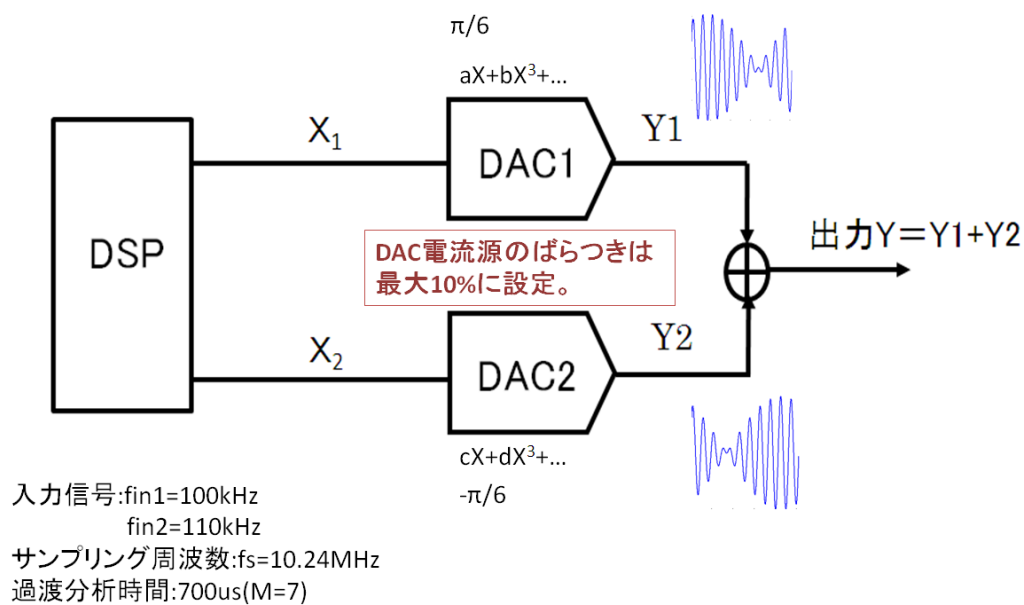
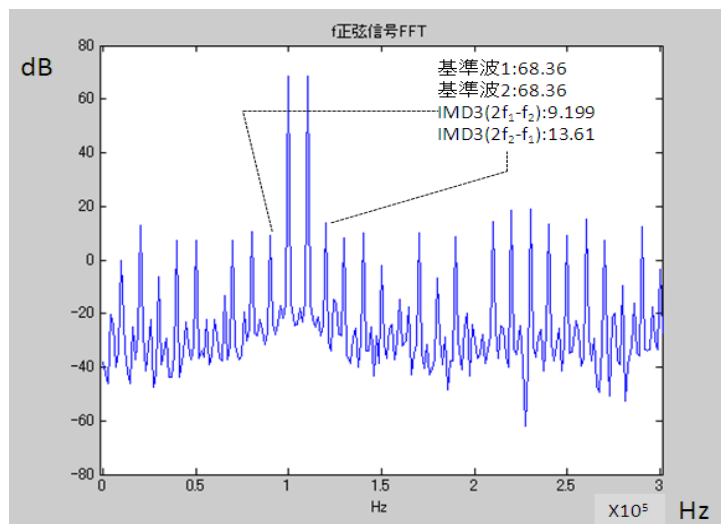


図 1.4.8 位相差合成型・2 トーン信号発生 IMD3 キャンセル

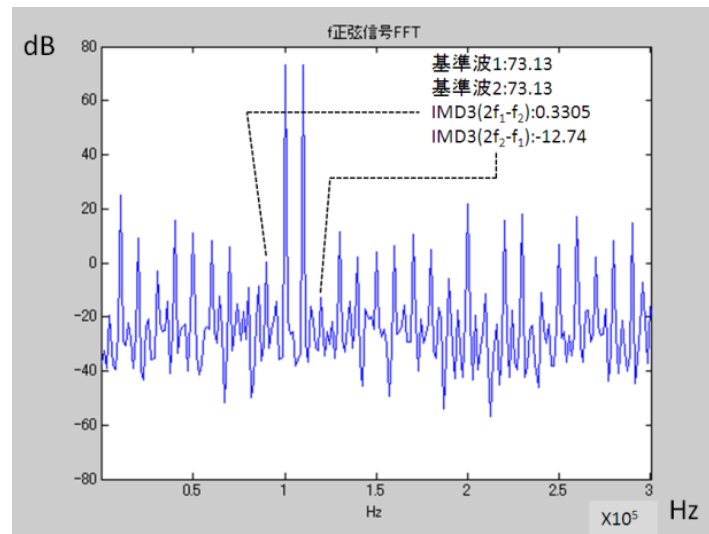
図 1.4.8 の条件に基づき、3 次歪みキャンセル (IMD3) 効果を SPICE のシミュレーションで確認した。

DAC 特性間にばらつきがない場合: $b = d$;

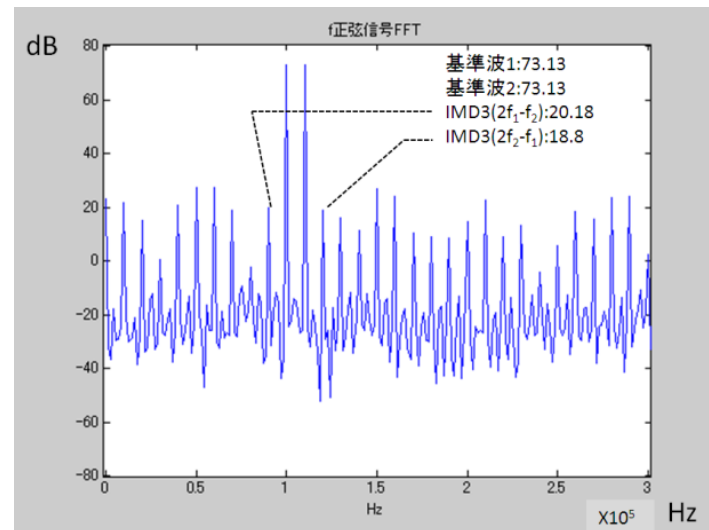
DAC 特性間にばらつきがある場合: $b \neq d$;



(a)従来手法



(b) 提案手法：DAC 特性間にばらつきがない場合



(c) 提案手法：DAC 特性間にばらつきがある場合

図 1.4.9 提案構成によるシミュレーション結果・2 トーン信号

図 1.4.9, 二つの DAC 中の電流セルの電流源のミスマッチが同じなら(c)のように従来手法(b)より IMD3 を減少できる。もし二つの DAC 中の電流セルと電流源のミスマッチが違うなら(d)のように IMD3 を低減できない。

1.5 ダイナミックエレメントマッチング（DEM）構成

上記図 1.4.5 と図 1.4.9 で DAC にばらつきがある場合は位相差信号合成型を適用しても歪成分をキャンセルできないことが確認された。ダイナミックエレメントマッチング構成（図 1.5.1）は上記の DAC 特性間にばらつきがある場合、三次歪みがキャンセルできない状況に対する改善提案である。

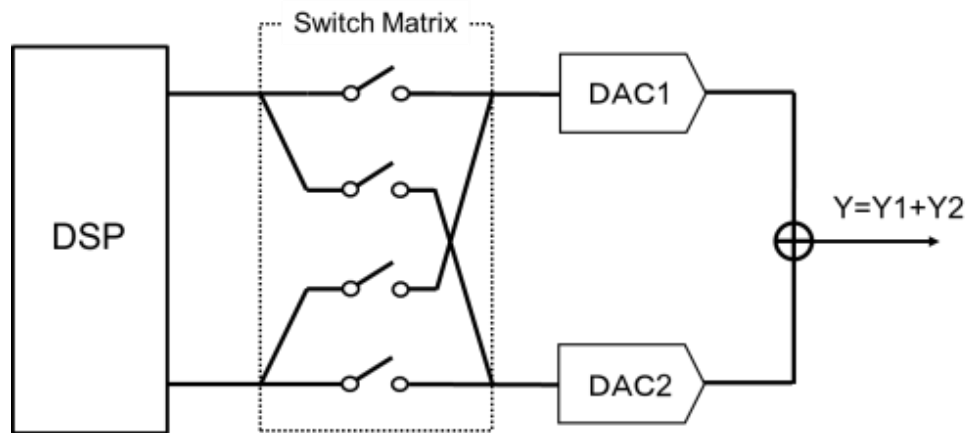


図 1.5.1 ダイナミックエレメントマッチング構成

このダイナミックエレメントマッチング構成は 1 クロック毎に信号の経路を変更することで DAC 特性のばらつきを時間平均化させる。1 クロックごとの経路の変化は図 1.5.2 に示す。

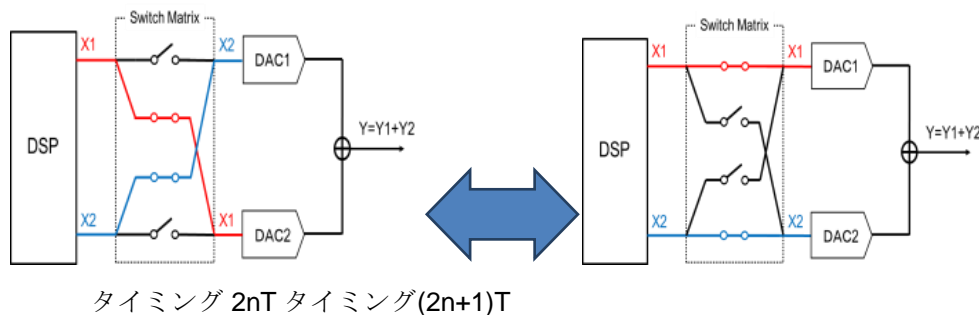
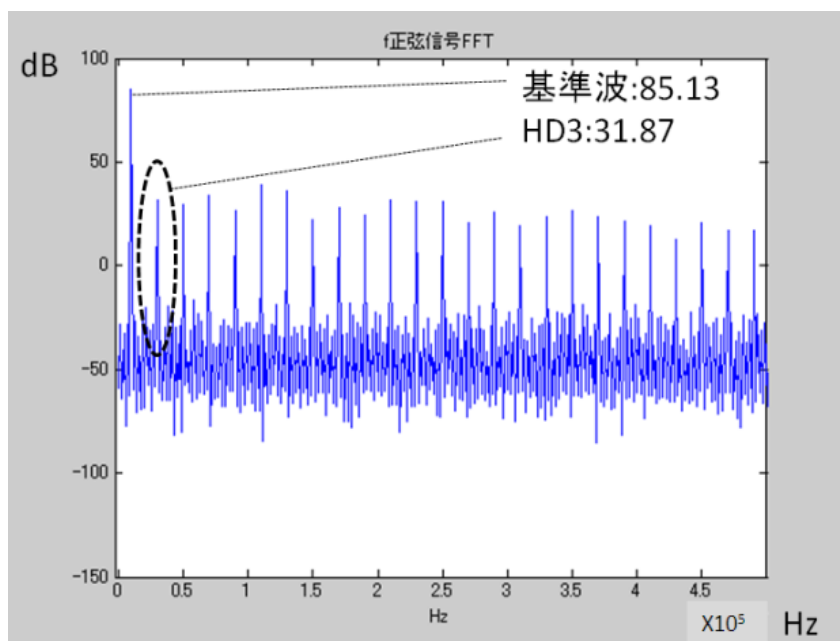


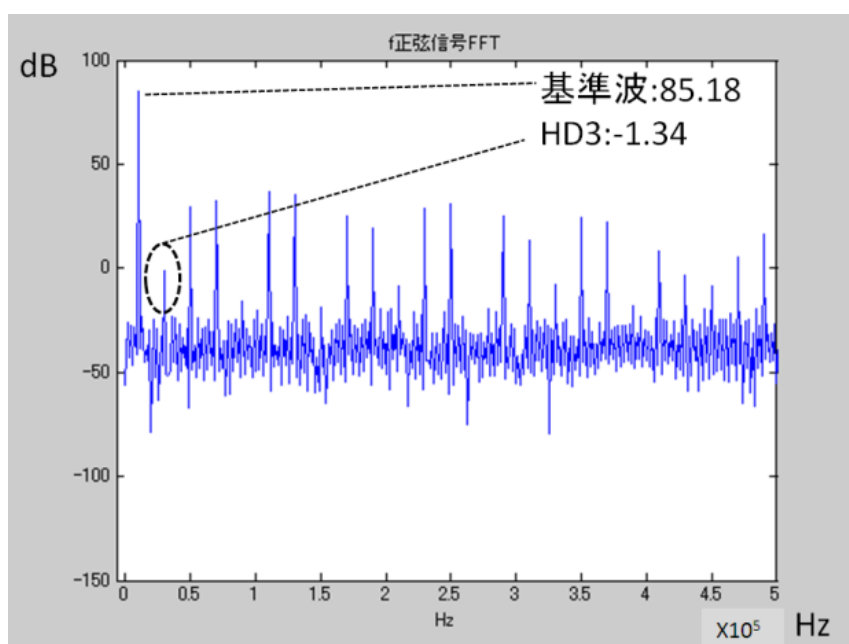
図 1.5.2 1 クロックごとの経路変化

ダイナミックエレメントマッチング構成の動作は最初のクロックでは上と下の 2 つのスイッチが ON する。それにより入力 X1 を DAC1、入力 X2 を DAC2 に入力する。次のクロックでは中央の 2 つのスイッチを ON にする。それによって入力 X1 を DAC2、入力 X2 を DAC1 に経路を切り替える。次のクロックでまた戻る。この切り替えを繰り返す。

次にダイナミックエレメントマッチング構成の効果を SPICE で検証を行った。



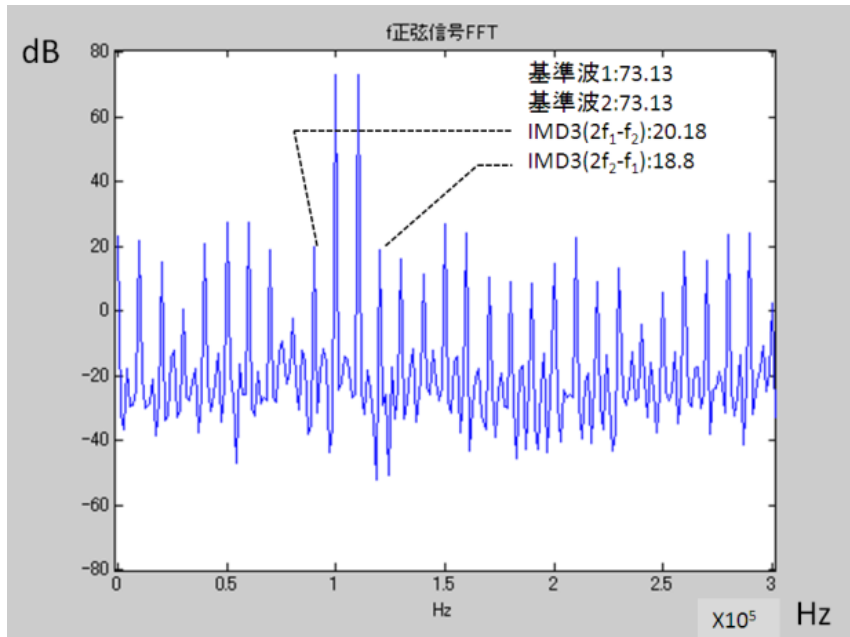
(a) 改善前(DEM 不使用)



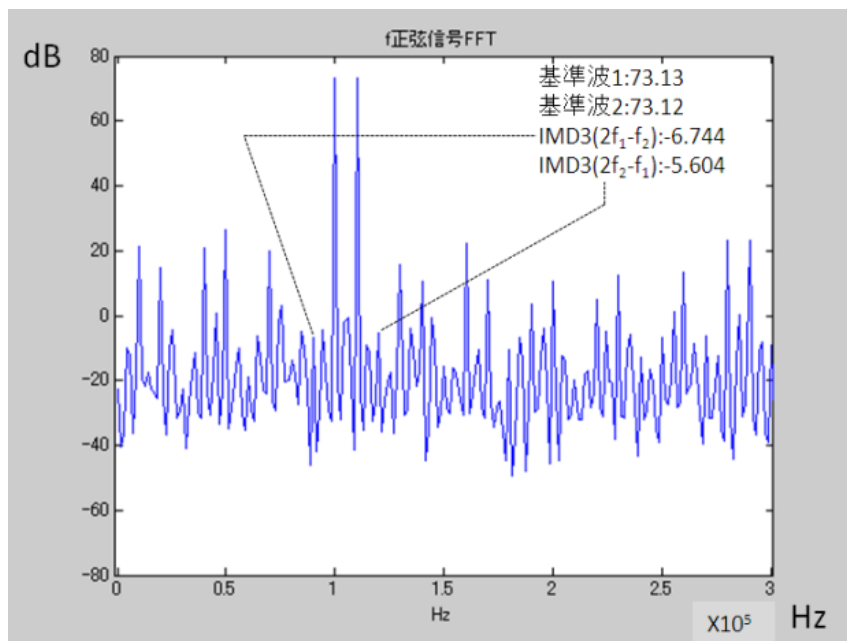
(b) 改善後(DEM 使用)

図 1.5.3 単一正弦波 @ DAC 特性間にばらつきがある場合

図 1.5.3 は位相差合成型が単一正弦波生成の場合 DEM 構成の使用前後の比較です。ダイナミックエレメントマッチング構成を用いることによって二つの DAC 特性間にばらつきがある場合でも発生信号の HD3 を減少できる。



(a) 改善前 (DEM 不使用)



(b) 改善後 (DEM 使用)

図 1.5.4 2 トーン信号 @ DAC 特性間にばらつきがある場合

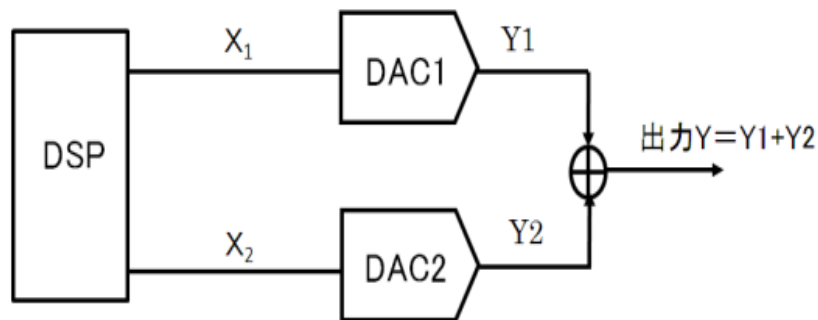
図 1.5.4 は位相差合成型が単一正弦波生成の場合 DEM 構成の使用前後の比較である。ダイナミックエレメントマッチング構成を用いることによって二つの DAC 特性間にばらつきがある場合でも発生信号の HD3 を減少できる。

1.6 位相差合成型+位相差切り替え型

1.6.1 「位相差合成型+位相差切り替え型」構造

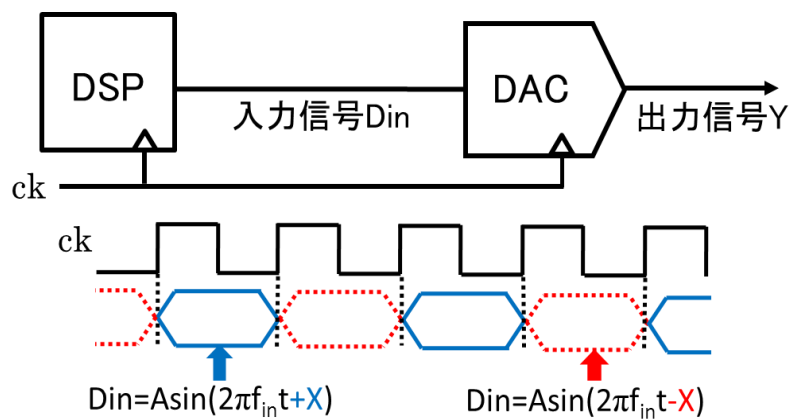
「位相差合成型+位相差切り替え型」は位相差合成型と位相差切り替え型で構成される。

位相差合成型



- メリット：高次歪みをキャンセルし、新しい成分の歪みをスペクトラムに持ち込まない。
- デメリット：3,5,7 次の歪みを同時にキャンセルと、8 個の DAC が必要。ハードウェア上の素子が増える。

位相差切り替え型[3][4][5]



- メリット：DAC を一つだけを用いて高次歪みをキャンセルできる。ハードウェア変更なしに DSP 部のプログラミングを変えるだけで実現できる。
- デメリット：
 - (1) サンプリング周波数によって新たな高次歪み成分が生成される。
 - (2) 新たな高次歪み成分がホワイトノイズと相互変調することでノイズフロアのレベルが増加する。

上記、二つの手法を組み合わせる提案するのは「位相差合成型+位相差切り替え型」である。二つの DAC を用いるだけで、3，5，7 次歪みを同時に減少することが可能である。

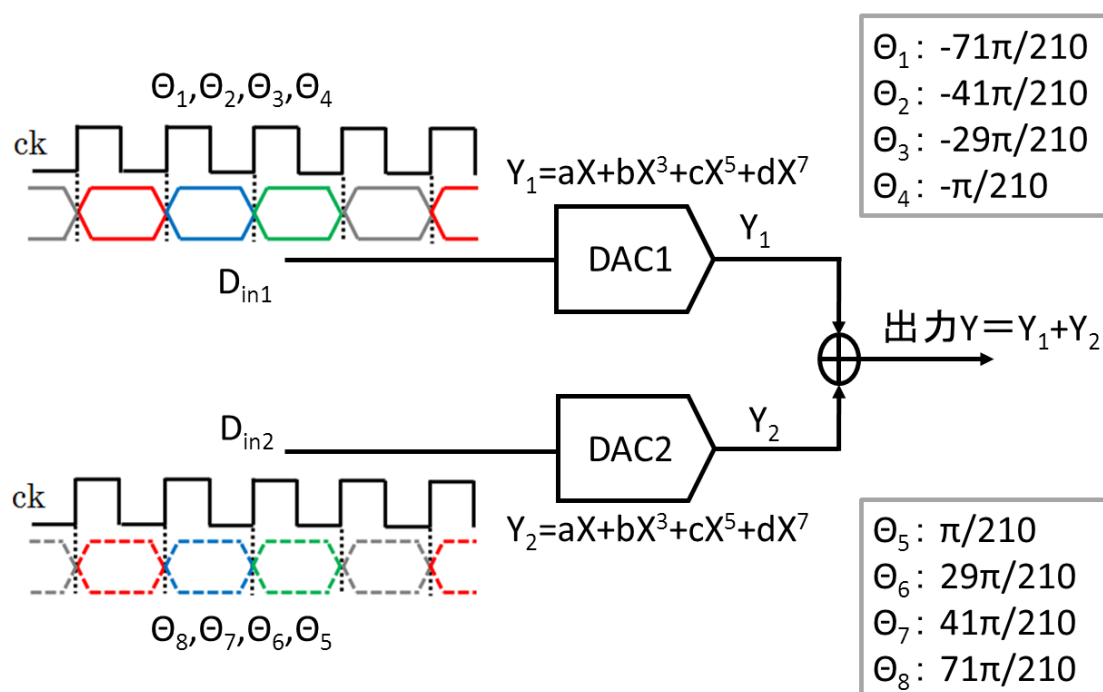


図 1.6.1 位相差合成型+位相差切り替え型

「位相差合成型+位相差切り替え型」の DAC 構成は位相差合成型と同じ二つの DAC を使って 3，5，7 次歪キャンセルによる位相のインターリーブ数が半分にできる。サンプリング周波数による歪み成分の影響も緩和できる。

※位相差切り替え型：DAC 一つに 8 個の位相を切り替え. ($\Theta_1\Theta_2\Theta_3\Theta_4\Theta_5\Theta_6\Theta_7\Theta_8$)

位相差合成型+位相差切り替え型：DAC 二つ、それぞれ 4 個の位相を順番通りに切り替える。

(DAC1: $\Theta_1\Theta_2\Theta_3\Theta_4$. DAC2: $\Theta_8\Theta_7\Theta_6\Theta_5$)

1.6.2 「位相差合成型+位相差切り替え型」原理

従来手法で DAC に 3,5,7 次歪がある AWG で単一正弦波を発生させる場合, 歪成分 HD3,HD5,HD7 が生成する(DACに3,5,7次非線形特性がある場合)。そこで二つDAC特性は $Y_{1,2}$ (式 1.6.1).入力信号は D_{in1} (式 1.6.2), D_{in2} (式 1.6.3).そして式(1.6.2)と式(1.6.3)を式(1.6.1)に代入し加算すると結果として HD3,HD5,HD7 成分が出でないことを確認できた。

$$Y_{1,2} = aD_{in} + bD_{in}^3 + cD_{in}^5 + dD_{in}^7 \quad \text{式 1.6.1}$$

$$D_{in1} = \begin{cases} D_{in10}(n) = A \sin(2\pi f_1 n T_s + \theta_1) & n = 4k \text{ の時} \\ D_{in11}(n) = A \sin(2\pi f_1 n T_s + \theta_2) & n = 4k + 1 \text{ の時} \\ D_{in12}(n) = A \sin(2\pi f_1 n T_s + \theta_3) & n = 4k + 2 \text{ の時} \\ D_{in13}(n) = A \sin(2\pi f_1 n T_s + \theta_4) & n = 4k + 3 \text{ の時} \end{cases} \quad \text{式 1.6.2}$$

$$D_{in2} = \begin{cases} D_{in20}(n) = A \sin(2\pi f_1 n T_s + \theta_8) & n = 4k \text{ の時} \\ D_{in21}(n) = A \sin(2\pi f_1 n T_s + \theta_7) & n = 4k + 1 \text{ の時} \\ D_{in22}(n) = A \sin(2\pi f_1 n T_s + \theta_6) & n = 4k + 2 \text{ の時} \\ D_{in23}(n) = A \sin(2\pi f_1 n T_s + \theta_5) & n = 4k + 3 \text{ の時} \end{cases} \quad \text{式 1.6.3}$$

位相条件：

$$\begin{cases} \theta_1 = -71\pi/210 \\ \theta_2 = -41\pi/210 \\ \theta_3 = -29\pi/210 \\ \theta_4 = -\pi/210 \end{cases} \quad \begin{cases} \theta_8 = 71\pi/210 \\ \theta_7 = 41\pi/210 \\ \theta_6 = 29\pi/210 \\ \theta_5 = \pi/210 \end{cases}$$

1.6.3 「位相差合成型+位相差切り替え型」 シミュレーション検証

そこで、ダイナミックエレメントマッチング（DEM）を投入して単一正弦波発生する場合のHD3、HD5、HD7 キャンセル効果を SPICE で検証した。

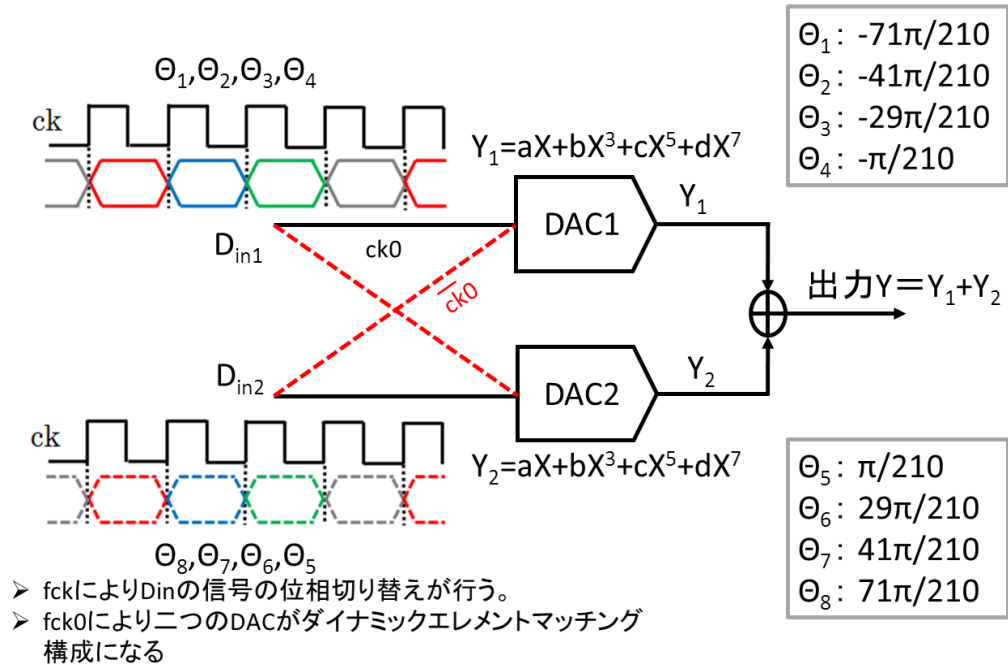
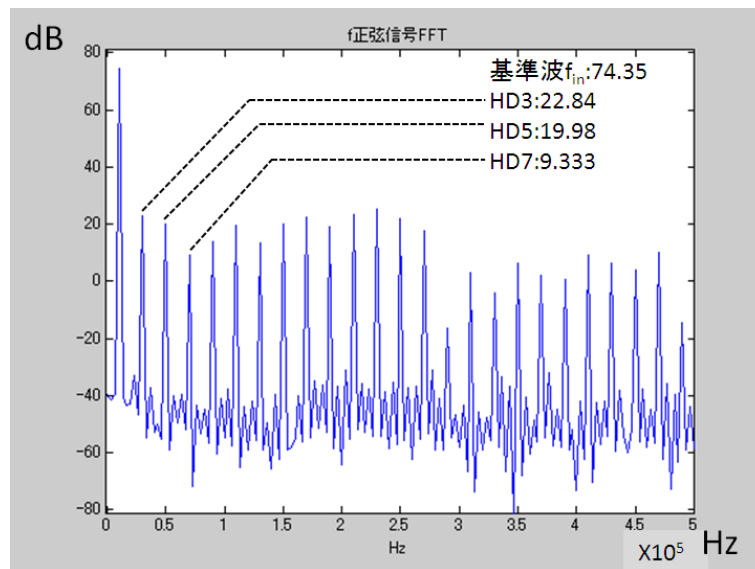


図 1.6.2 DEM 投入した位相差合成型+位相差切り替え型

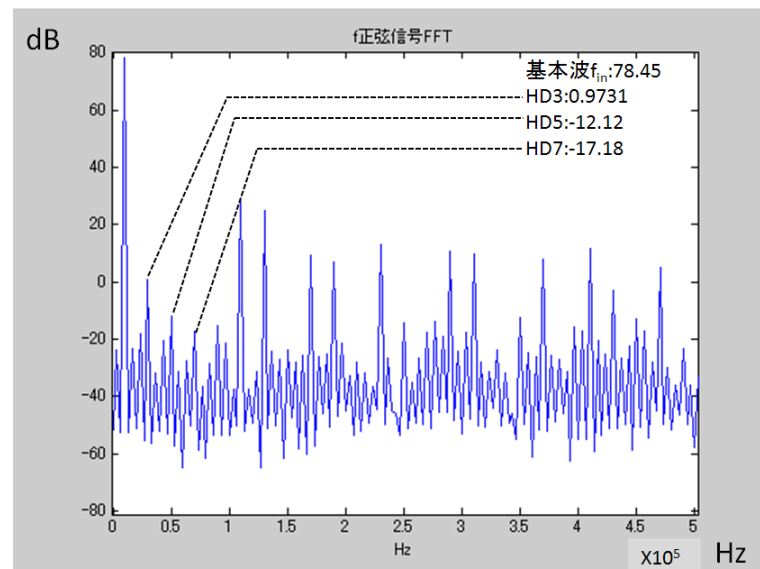
基本波	10kHz
f_{ck}	102.4MHz
f_{ck0}	409.6MHz

表 1.6.1 シミュレーション条件

表 1.6.1 のシミュレーション条件に基づき、検証を行なった。



(a) 従来手法



(b) 位相差合成型+位相差切り替え

図 1.6.3 シミュレーション結果

図 1.6.3 提案手法(b)「位相差合成型+位相差切り替え」を用いた二つの DAC 線形性特性がばらつく場合でのダイナミックエレメントマッチング構成で生成した単一正弦波の出力スペクトラムである。この手法を用いる事で HD3、HD5、HD7 のキャンセル効果を確認することができた。

1.7 ま と め

- 従来手法では3次歪みがある場合に単一正弦波(2 トーン信号)を生成すると出力に歪み成分 HD3(IMD3) が発生するが DAC を 2 つと加算器を使用すれば HD3(IMD3) のキャンセルが可能である。
- 位相差合成型の二つの DAC 特性間にばらつきがない場合,三次歪みが低減できるが二つ DAC 特性間にばらつきがある時に歪みの低減ができないことを SPICE 上で確認できた。
- ダイナミックエレメントマッチング構成を用いることによって二つ DAC 特性間にばらつきがあっても 3 次歪みを減少できた。
- 「位相差合成型+位相差切り替え型」を用いることによって, 二つの DAC を用いるだけで, 3, 5, 7 次歪みを同時に減少することが可能である。

第二章 電源回路高効率化の研究

2.1 研究背景と目的

2.1.1 サーバ電源の高効率要求

近年、環境への関心が高まり、「グリーン IT」というコンセプトを取り入れての電源設計が行なわれている。データセンタにおける機器やシステムの省エネ化、省資源化が課題になり特にその中心となるサーバの省電力化が求められている。

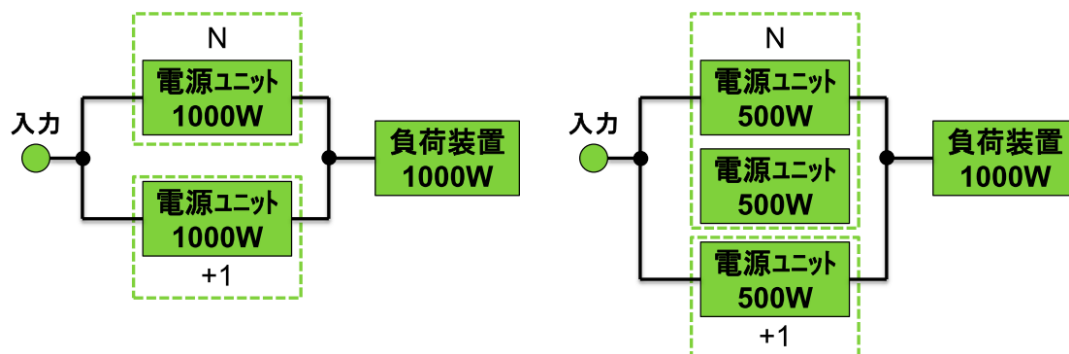


図 2.1.1 N+1 運転方式

図 2.1.1 が示したように N+1 冗長運転とは電源ユニットを 2 台以上並列に接続して装置に電力を供給することで 1 台の電源ユニットが故障しても残りの電源で装置に電力を供給し続けることができる運転方法である。例えば負荷が 1000W 必要であれば出力容量 1000W の電源なら 1 台、出力 500W の電源なら 2 台を並列にすれば負荷の装置は動作するが、さらにもう 1 台 (+1) つなぐことによって 1 台が故障しても動作を継続することを可能とした運転方式である。サーバなどではシステムの信頼性を高めるため、このような電源構成が取られている。

その N+1 冗長運転方式について 80PLUS プログラム (www.80plus.org) が推進する電気機器の省電力化プログラムはコンピュータやサーバの電源が 10%、20%、50%、100%の負荷環境化において電源変換率が 80%以上の基準を満たした製品を表 2.1.1 のようにランク分けしている。



% of Rated Load	10%	20%	50%	100%
BRONZE	N/A	80%	85%	81%
SILVER	N/A	85%	89%	85%
GOLD	N/A	88%	92%	88%
PLANTINUM	N/A	90%	94%	91%
TITANIUM	90%	94%	96%	91%

表 2.1.1 80 PLUS Certification

2.1.2 電源回路の効率劣化原因

スイッチング電源回路は PFC(AC/DC)回路と DC/DC コンバータ回路で構成されている。(図 2.2.1)

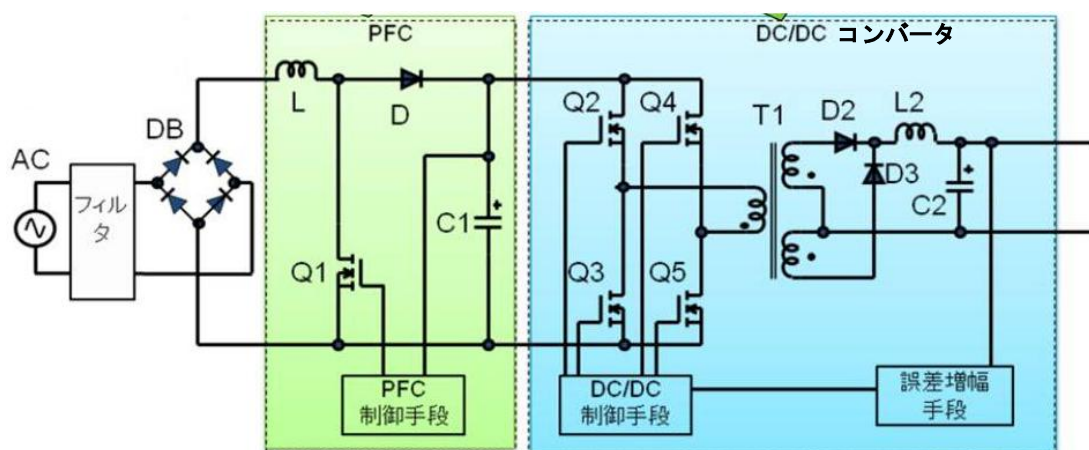


図 2.1.2 スイッチング電源の構成

PFC（Power Factor Correction）回路とは「力率改善回路」のことであり、力率を1に近づけ高調波電流の発生を抑制する整流・平滑回路である。

DC/DC コンバータ回路は、PFC 回路により整流平滑された電源電圧を絶縁して安定した直流電圧に変換する機能がある。

上記のスイッチング電源回路には効率を劣化させる原因が主に二つある。一般的に PFC 回路では、昇圧比が低く出力電圧が低いほど高効率を達成することができる。しかし、ワイドレンジの電源回路では入力電圧範囲は AC 90～265V であり、この場合は AC 265V のピーク値 ($265\sqrt{2}=375\text{V}$)以上の高い電圧(390V)がリンク電圧になってしまう。これが効率の悪化の原因になっている。また、DC-DC コンバータ回路は PWM 制御で常に固定周波数で MOSFET がスイッチングしており、このために軽負荷時に効率が大幅に劣化する。これが二つ目の原因である。

2.1.3 研究目標

サーバ電源はN + 1 冗長運転方式で構成されている。したがって、電源回路はフル負荷で動作することが稀で通常 20%~50% 負荷で動作しているために、軽負荷の電源効率を向上させることが重要である。しかしながら、軽負荷の場合に上記の効率劣化原因を克服し高い効率を得るためには、回路トポロジーの改良だけでは実現困難である。そこで、我々は回路トポロジーを変えずに DSP を用いたデジタル制御を駆使し、最適制御でハーフロードまたは 20% 近くの軽負荷領域での効率の向上方法（プログラムの開発）を検討する。

電源回路の検討は二つの回路に分けて行なった。

BLPFC AC/DC 回路 (Bridgeless Power Factor Correction AC/DC)

- リンク電圧の最適可変 (@ 50% Load)
- PWM (スイッチング周波数) の最適可変 (@ 5%~20% Load)

PSFB DC/DC 回路 (Phase Shift Full Bridge DC/DC)

- PWM (スイッチング周波数) の最適可変 (@ 10%~20% Load)

2.1.4 研究環境

TI 社が提供した開発ソフト CCS——Code Composer Studio (図 2.1.3) を用いて、研究を行なった。

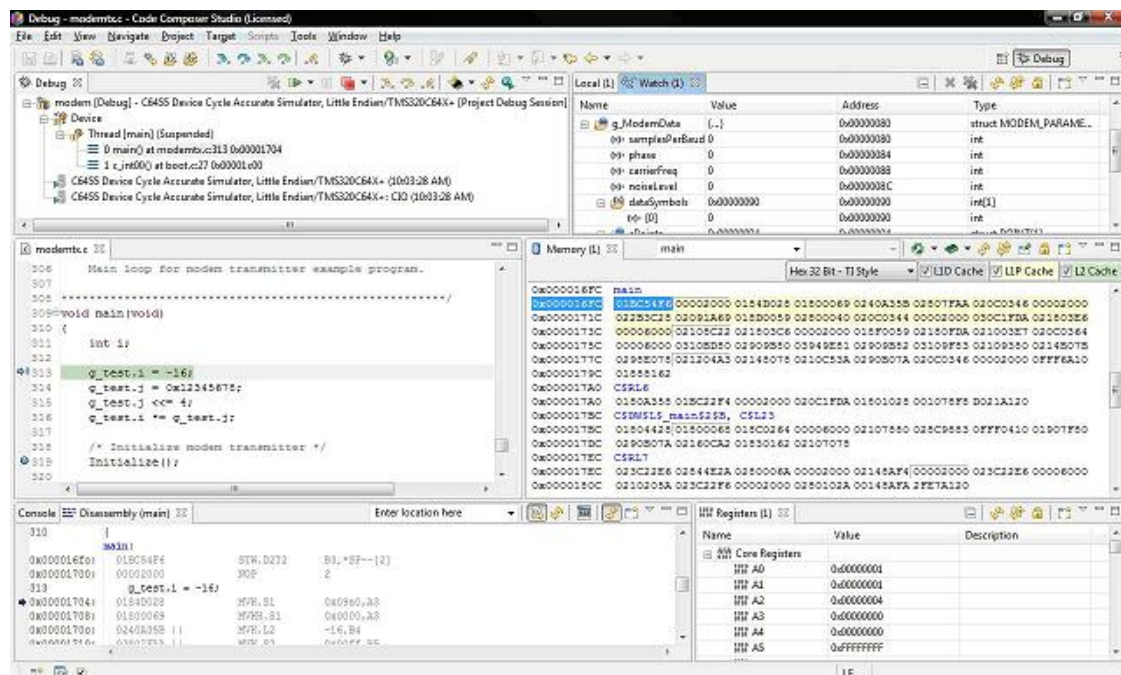


図 2.1.3

Code Composer Studio™ (CCS) は、テキサス・インスツルメンツ (TI) の組込みプロセッサ向けの統合開発環境 (IDE) である。CCS は、組込みアプリケーションの開発およびデバッグに必要なツールで構成され TI の各デバイス・ファミリ向けのコンパイラ、ソース・コード・エディタ、プロジェクト・ビルド環境、デバッガ、プロファイラ、シミュレータ、リアルタイム・オペレーティング・システムなど多数の機能が含まれている。IDE にはアプリケーションの開発フローをステップごとに実行できる単一のユーザー・インターフェイスが備わっている。使い慣れたツールとインターフェイスを使用することでユーザーはすぐに開発を開始し、高度で生産性のよいツールを利用してアプリケーションに機能を追加することが可能である。

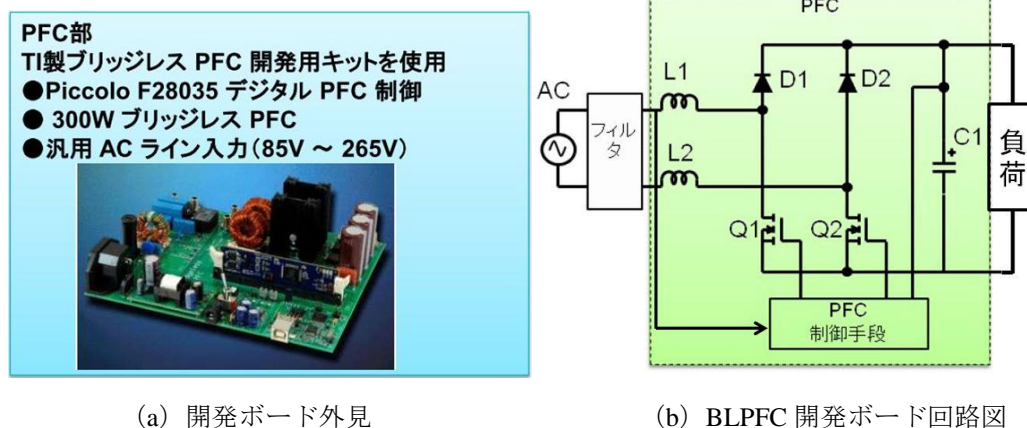
2.2 BLPFC AC/DC 回路 (Bridgeless Power Factor Correction AC/DC)

2.2.1 BLPFC AC/DC 回路開発ツール

実験は TI 社製のブリッジレス PFC 開発ボードを使用した。(図 2.2.1)Piccolo F28035 (DSP) で PFC AC/DC を制御している。回路の基本仕様は下記に示す。

- Input Voltage(AC line):
95V(Min)to 250V(Max),47~63Hz
- **400Vdc Output**
- 300 Watts Output Power
- Full Load efficiency greater than 93%.
- Power factor at 50% or greater load – 0.98(Min)
- **PWM frequency 200kHz**

上記のように、リンク電圧と PWM 周波数が規定されている。



2.2.2 BLPFC AC/DC 回路——リンク電圧の最適可変(@ 50% Load)

電源回路はリンク電圧が常に最大入力電圧のピーク値(390V~400V)になるように制御しており、それが効率の悪化の原因である。その解決方法として、DSP が入力電圧の実効値をモニタし、最適の昇圧比を決めればリンク電圧を最適値にすることが可能と考え、それに基づきプログラムを開発した。

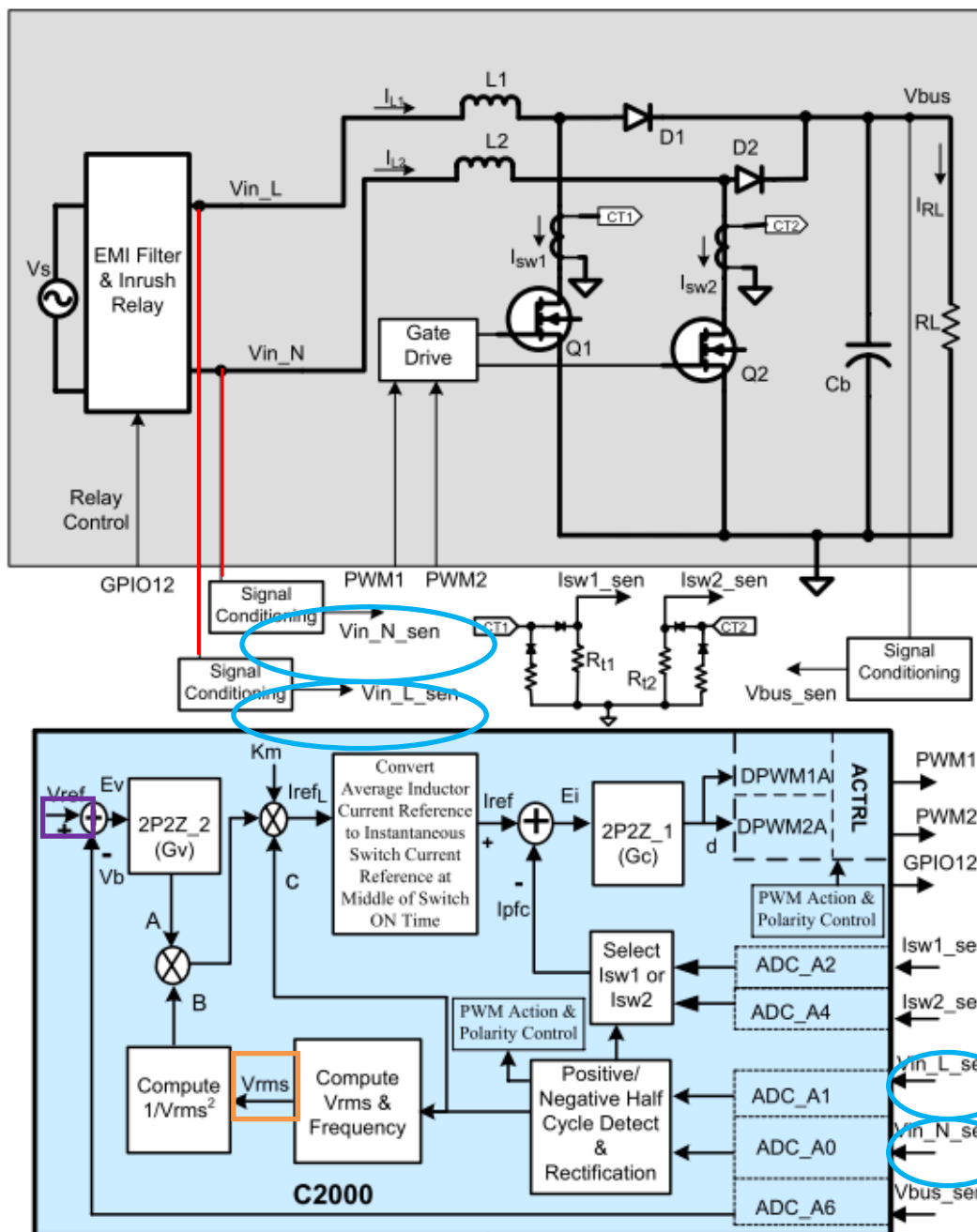


図 2.2.2 ridgeless PFC Converter Control using C2000 Micro-controller

STEP1:入力電圧の検出(V_{inac})

STEP2:入力電圧を AD 変換器によって、DSP に取り込む

STEP3:入力電圧の実効値を算出(V_{rms})

STEP4:リンク電圧の目標値 (V_{ref}) を下記の式で制御する

$$\text{リンク電圧} = \text{最適昇圧比} \times V_{rms}$$

上記のステップに基づきプログラムを作り実験を行なった。

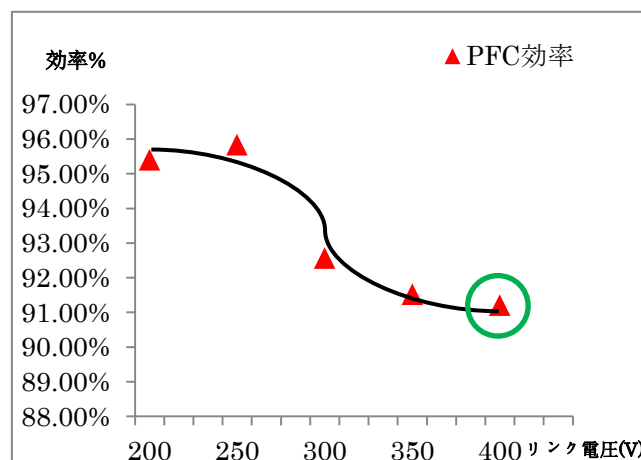


図 2.2.3 それぞれのリンク電圧に対応するハーフロードの PFC ボード出力効率

仕様に規定されたリンク電圧は 400V である。ハーフロードの場合では図 3 で示したように 400V のリンク電圧で最高の効率を得られなかったのが最適ではない。一方、昇圧比を下げる（リンク電圧を下げる）[こと](#)によってハーフロードでの効率向上を確認できた。

2.2.3 PFC AC/DC 回路——PWM(スイッチング周波数)の最適可変 (@ 5%~20% Load)

軽負荷の場合、電源回路は固定周波数の PWM 制御での MOSFET のスイッチングロスにより効率が大幅に劣化する。そこでデジタル制御で PWM スwitching 周波数を可変にし、PFC 回路の効率向上を検討した。

プログラムを開発する前に異なる PWM 周波数に対応する軽負荷の場合の PFC AC/DC ボード効率を計った。仕様に規定されたスイッチング周波数は 200kHz である。軽負荷の場合では図 2.2.4 が示したように 200kHz で PFC AC/DC ボードでの高効率が得られないので最適周波数ではない。

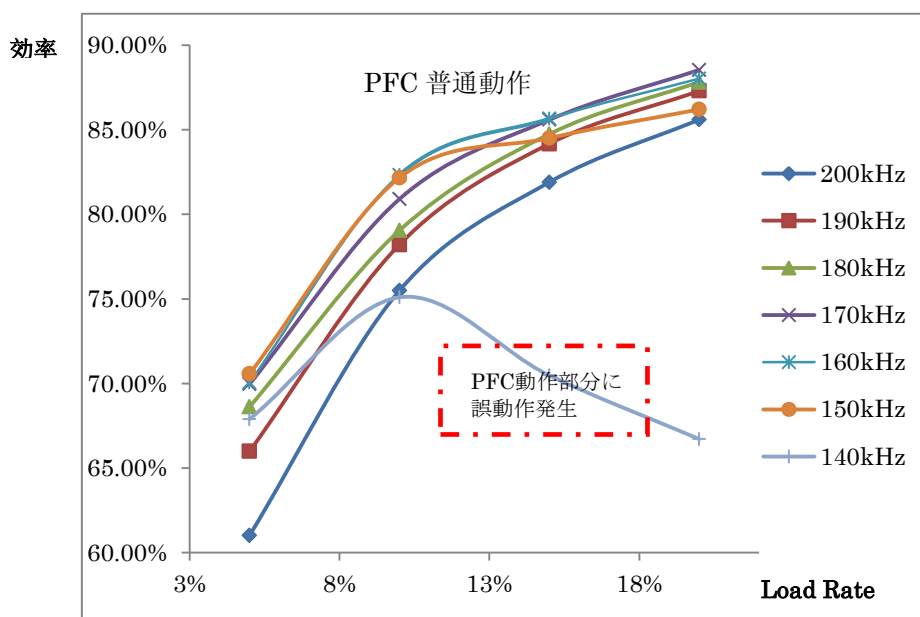
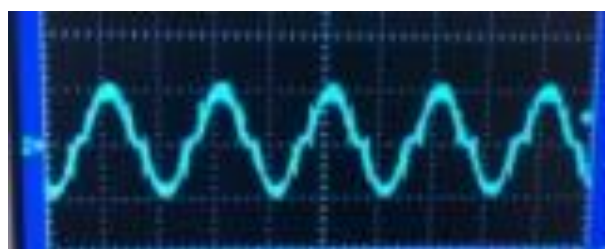
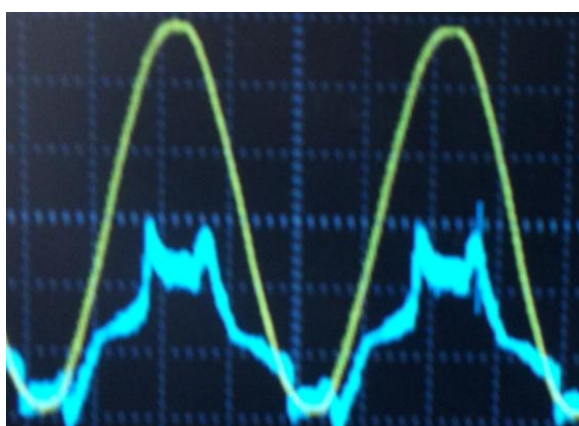


図 2.2.4 異なる PWM 周波数に対応する軽負荷の場合の PFC AC/DC ボード効率
 140kHz に PFC 動作部分に誤動作が発生した。図 2.2.5 の(b)が示したように青い入力電流は黄色
 入力電圧 (50Hz) 波形に比べると PWM 周波数が低すぎる場合 PFC 動作部分が誤動作する (電
 流波形が崩れる) ので回路効率が大きく劣化し、出力電圧も不安定になる。



(a) PFC 正常動作の入力電流波形



(b) PFC 動作部分に誤動作発生

図 2.2.5

図 2.2.4 に基づき、軽負荷の場合の PWM 周波数範囲を表 2.2.1 で示す。

Load Rate(%)	PWM 周波数(kHz)
5% - 10%	150kHz
10% - 15%	160kHz
15% - 20%	170kHz

表 2.2.1 PFC AC/DC ボード最適 PWM 稼働範囲(@軽負荷)

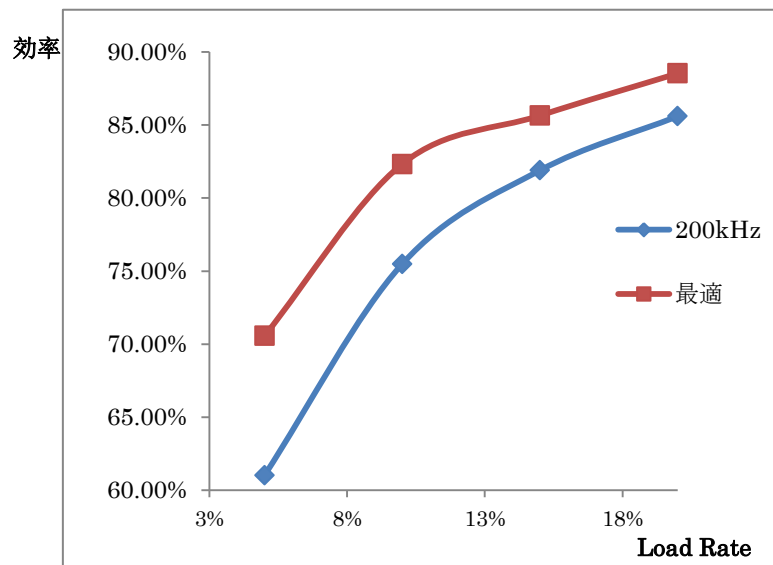


図 2.2.6 固定周波数の場合と周波数可変の場合の PFC 回路の効率比較

2.3 PSFB DC/DC 回路 (Phase Shift Full Bridge DC/DC Converter)

2.3.1 PSFB DC/DC 回路開発ツール

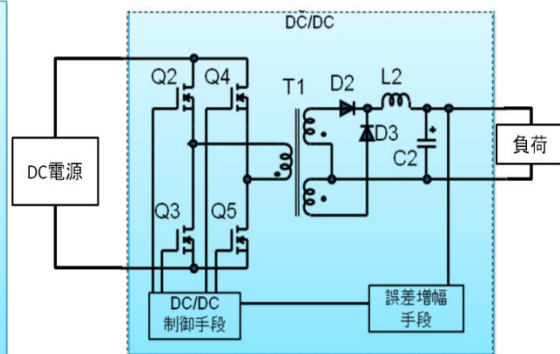
実験はTI製のブリッジレスPFC開発ボードを使用した。TMS320F28027 (DSP) でPSFB DC/DC を制御している。回路の基本仕様は次に示す。

- 400V DC input(370Vdc to 410Vdc operation),
12V DC output
- Peak efficiency greater than 95%
- 50A(600Watt)rated output
- Phase Shifted Full-Bridge Circuit topology
- **100kHz switching frequency**

上記のように、PWM 周波数が規定されている。



a) 開発ボード外見



(b) PSFB DC/DC 開発ボード回路図

図 2.3.1 TI 製デジタル制御フェーズ・シフト・フル・ブリッジ DC/DC コンバータ開発用ボード

2.3.2 PSFB DC/DC 回路——PWM(スイッチング周波数)最適可変(@10%~20% Load)

軽負荷の場合、固定周波数での MOSFET のスイッチングロスにより効率が大幅に劣化する。そこでデジタル制御でスイッチング周波数を可変することで PSFB DC/DC 回路の効率向上を検討した。

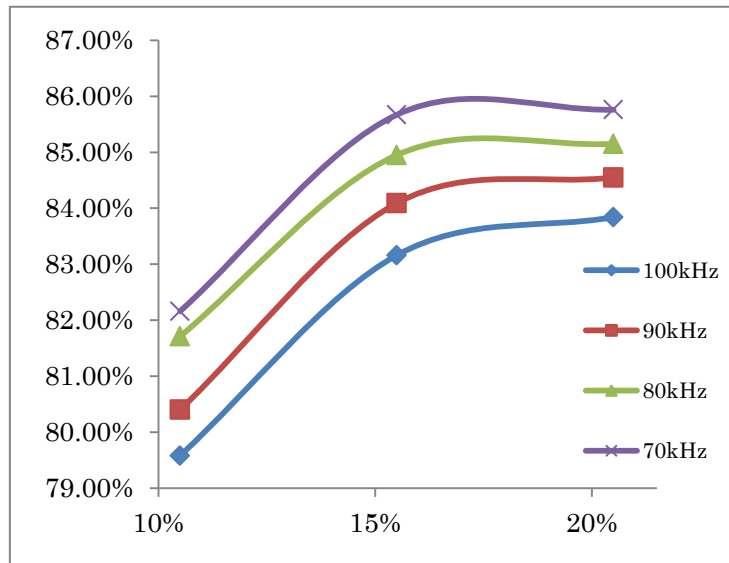
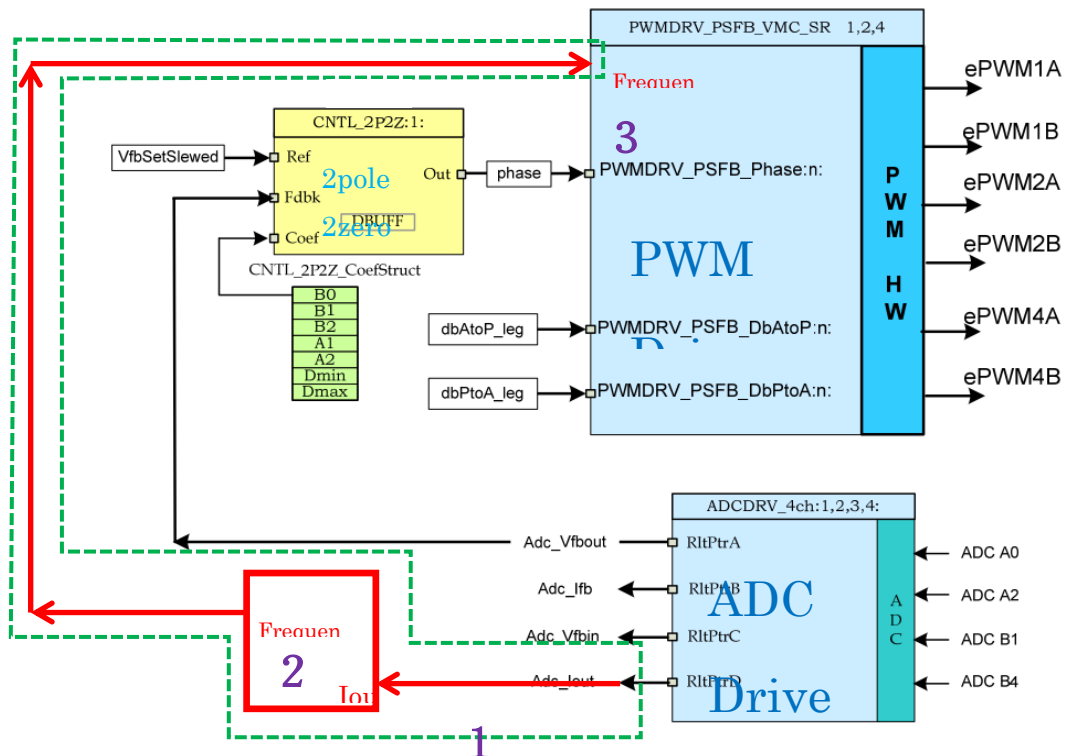
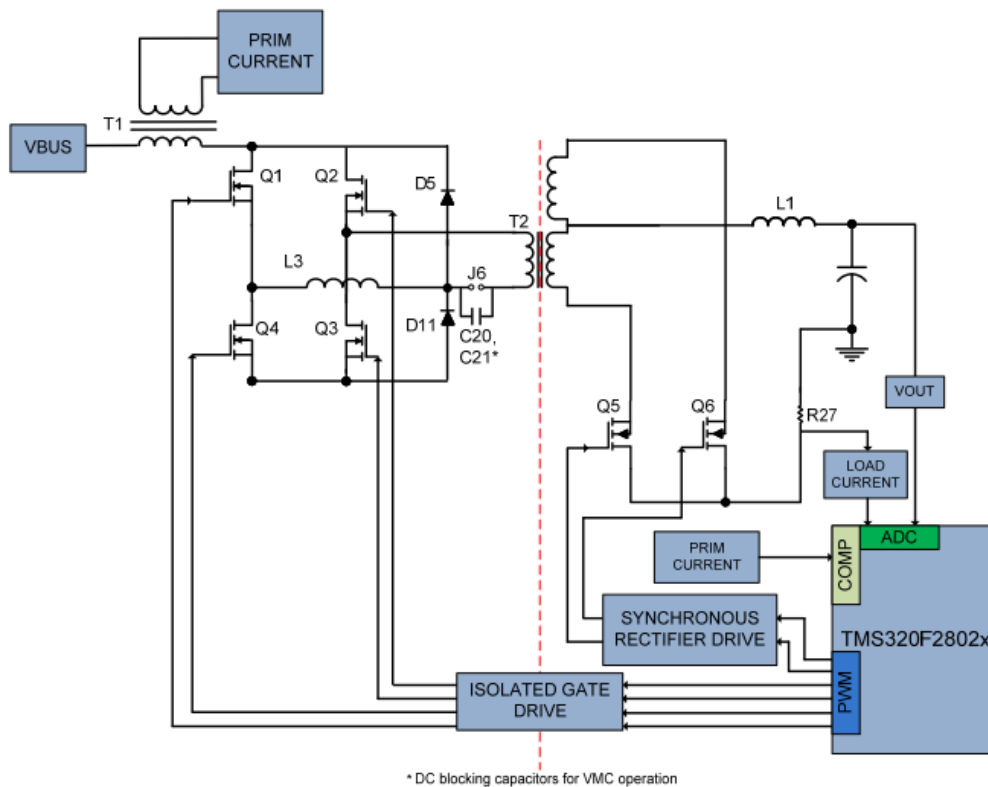


図 2.3.1 異なる周波数に対する、軽負荷時の PSFB DC/DC ボード効率

仕様に規定されたスイッチング周波数は 100kHz である。軽負荷の場合では、図 2.3.1 が示すようにスイッチング周波数 100kHz は最適ではない。



(a)



(b)

図 2.3.2 PSFB DC/DC VMC Control Flow and Circuit topology

図 2.3.2(a)のコントロールフローチャットに各モジュールに対応するシグナルの意味は下記に示す。

PWM モジュールスイッチングコントロール信号

Primary side

- ePWM1A PWM drive for full-bridge switch Q1
- ePWM1B PWM drive for full-bridge switch Q4
- ePWM2A PWM drive for full-bridge switch Q2
- ePWM2B PWM drive for full-bridge switch Q3

Second side

- ePWM4A PWM drive for synchronous rectifier switch Q5
- ePWM4B PWM drive for synchronous rectifierswitch Q6

フェードバック信号

- Adc_Vfbout PSFB output voltage
- Adc_Ifb Transformer primary current
- Adc_Iout PSFB output current
- Adc_Vfb PSFB input voltage

図 2.3.2(a)赤い部分は PWM スイッチング周波数を可変にするプログラムとしてシステムに付け加えるモジュールであり、その機能は下記のようなになる。

STEP1:出力電流 Iout のモニタ

STEP2: 出力電流大きさ（電力）によるスイッチング周波数の確定

STEP3: PWM Drive Module の周波数パラメータを更新

現段階では、PSFB DC/DC ボードの周波数を可変にすることに成功したが最適の制御が出来ていない。したがって今後の実験に備え周波数可変のプログラムでの予備実験を行なった。予備実験の内容は表 2.3.1 に基づき PSFB DC-DC ボードで出力電流（電力）をモニタし、スイッチング周波数は電流の大きさに応じて自動的に変化するプログラムを開発した。

(Full Range : 50Amp (Full Load))

出力電流 (A)	スイッチング周波数 (k H z)
0 ～ 5	6 0
4 ～ 8	6 5
7 ～ 1 1	7 0
1 0 ～ 1 4	8 0
1 3 ～ 1 7	9 0
1 6 ～	1 0 0

表 2.3.1 異なる出力電流に対応するスイッチング周波数@軽負荷

図 2.3.1 が示した予備実験の仕様は下記のフローチャートでプログラム化し、実験を行なった。結果として、PSFB DC/DC 開発ボードが表 2.3.1 の仕様通りで動作した。(付録)

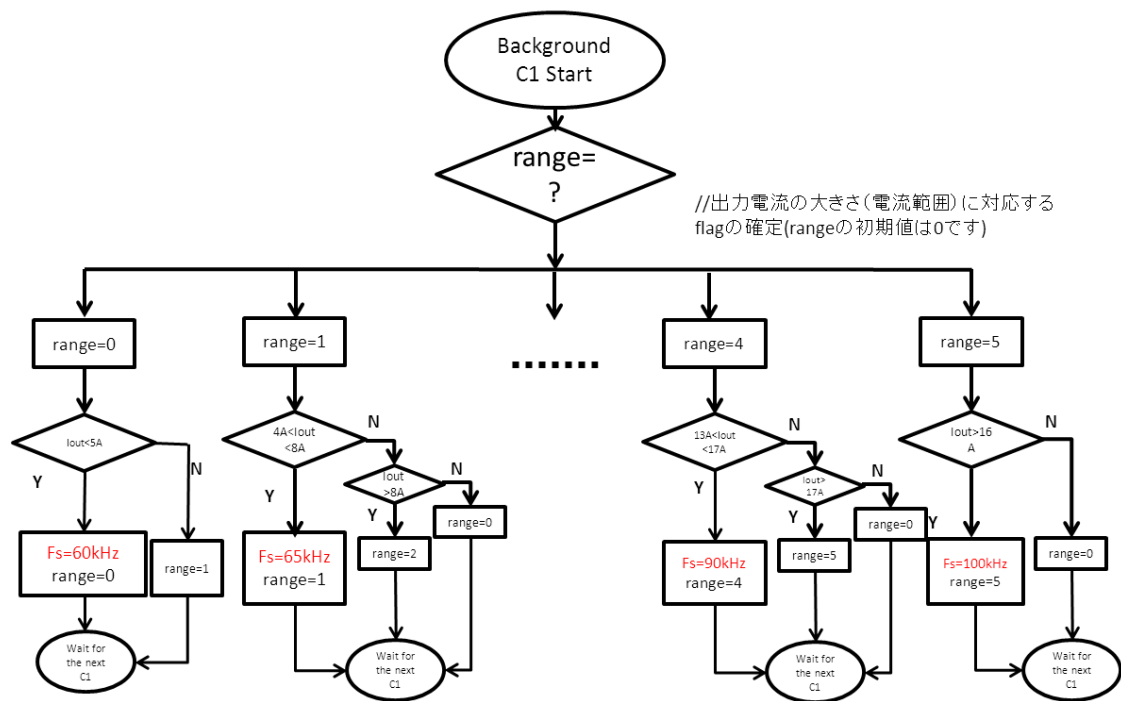


図 2.3.3 フローチャート

2.4 ま と め

本研究は TI 社の C2000 シリーズの DSP を用いて電源回路の軽負荷場合の効率向上方法を検討し、その指針を得た。

DSP や CPU の製造コストが低減し、これからの電源回路をプロセスコントローラで制御するのが主流になっていくと考えられる。本研究を通じて DSP デジタル制御のメリットは以下のようになることが判明した。

- 柔軟性 (Flexible)

電源回路のハードウェアを変えずにソフトウェア上でリンク電圧と PWM 周波数を変えることが容易に実現できる。軽負荷の場合 PWM と PFM 制御を切り替える必要はなく、ハードウェア構成が簡素化できる。

- 可視化 (Visible)

リンク電圧や PWM 周波数の可変部分をソフトウェア上で関数化し、モジュールとしてシステムに付け加えることができる。それらの制御関数は式として明示でき、修正や変更も容易に可能である。

本研究で解決すべき問題点。

- BLPFC AC/DC 開発ボードが低周波数で動作する場合の PFC 誤動作の原因が解明できていない。
- 本研究に BLPFC AC/DC ボードの入力電圧は 100V であるが、高い入力電圧(250V~265V)での効率向上効果を未確認である。
- 二つの開発ボードにテストピンをつけて回路動作の分析を行なう予定である。
- データ数が少ないために分析の精密性を欠いている。これから実験でデータ数を多く取る予定である。

謝辞

本論文を作成するにあたり、指導教官の群馬大学大学院工学研究科小林春夫教授から二年間に渡り丁寧且つ熱心なご指導を賜りました。ここに深く感謝の意を表します。

また群馬大学、高井伸和準教授、落合政司客員教授、小堀康功客員教授には有益な助言を頂き、心より感謝いたします。研究室、研究生活面でお世話になりました石川信宣技官に感謝いたします。

群馬大学小林研究室、加藤啓介、安部文隆、須釜裕太、劉羽、易茹、ナスカー・ザカリー、靳光磊、李慕容、呉澍、諸氏の有益な助言と議論をして頂き本研究を行う上で大変良い刺激になりました、ここに感謝いたします。

本研究のアイデアを下さり大変有意義なご討論をいただきました(株)STARC 小林修氏、(株)サンケン電気鈴木庸弘氏、麻生真司氏に心より感謝するとともに厚く御礼申し上げます。

最後に、本研究に対して沢山の有益な機会を与えて下さった関係者の皆様に心より感謝申し上げます。

発表論文

- ①高川、若林和行、加藤啓介、安部文隆、小林春夫、小林修、松浦達治、新津葵一、高井伸和「低歪み信号発生用 DA 変換器アーキテクチャ」電気学会 電子回路研究会、ECT-12-020、横須賀 (2012 年 3 月 29 日)
- ② Keisuke KATO, Fumitaka ABE, Kazuyuki WAKABAYASHI, Chuan GAO, Takafumi YAMADA, Haruo KOBAYASHI, Osamu KOBAYASHI, Kiichi NIITSU “Two-Tone Signal Generation for Communication Application ADC Testing”, The 21st IEEE Asian Test Symposium, Niigata, Japan (Nov. 2012).
- ③ Guanglei Jin, Hao Chen, ChuanGao, YunpengZhangHaruo Kobayashi, NobukazuTakai, KiichiNiitsu, KhayrollahHadidi, "Digitally-Contolled Gm-C Bandpass Filter", IEEE Asia Pacific Conference on Circuits and Systems, Kaohsiung, Taiwan (Dec. 2012).
- ④ K. Kato, F. Abe, K. Wakabayashi, C. Gao, T. Yamada, H. Kobayashi, O. Kobayashi, K. Niitsu “Two-Tone Signal Generation for Testing of Communication Application Devices”, 第 25 回回路とシステムワークショップ、淡路島 (2012 年 7 月 30 日) .
- ⑤ 靳 光磊、陳昊、高川、張雲鵬、小林春夫、高井伸和、新津葵一「連続時間アナログフィルタのデジタル自動調整法の一提案」電気学会 電子回路研究会、ECT-12-038、横須賀 (2012 年 3 月 30 日)
- ⑥高川、ジン・コウライ*, 李慕容 (群馬大学), 落合政司, 鈴木庸弘, 麻生真司 (サンケン電気), 小堀康功, 小林春夫, 高井伸和, 志水勲 (群馬大学)「C2000 シリーズ DSP 用いたスイッチング電源回路軽負荷場合の効率向上手法の検討」電気学会東京支部栃木・群馬支所合同研究発表会 ETT-12-5 (2013 年 2 月 28 日発表予定)

参考文献

- [1] 小林春夫,山口隆弘「ディジタルアシスト・アナログテスト技術」電子情報通信学会集積回路研究会, 大阪 (2010 年 7 月)
- [2] 本木義人, 菅原秀武, 小林春夫, 小室貴紀, 酒寄寛, 「通信用 AD 変換器テスト評価のためのマルチトーン・カーブ・フィッティング・アルゴリズム,電子情報通信学会和文誌 C,vol.J86-C,no.2,pp.186-196 (2003 年 2 月).
- [3] 若林和行, 上森聡史, 山田貴文, 小林修, 加藤啓介, 小林春夫, 新津葵一, 松浦達治「ADC テスト信号生成のための AWG 非線形補正技術」第 6 4 回 FTC 研究会岐阜 (2011 年 1 月)
- [4] 安部文隆, 加藤啓介, 若林和行, 小林修, 小林春夫, 新津葵一「インターリーブを用いた低歪み 2 トーン信号発生技術」電気学会電子回路研究会長崎(2011 年 10 月 20 日, 21 日)
- [5] K. Wakabayashi, T. Yamada, S. Uemori, O. Kobayashi, K. Kato, H. Kobayashi, K. Niitsu, H. Miyashita, S. Kishigami, K. Rikino, Y. Yano, T. Gake, "Low-Distortion Single-Tone and Two-Tone Sinewave Generation Algorithms Using an Arbitrary Waveform Generator", IEEE International Mixed-Signals, Sensors, and Systems Test Workshop, Santa Barbara, CA (May 2011)
- [6] T. Yamada, O. Kobayashi, K. Kato, K. Wakabayashi, H. Kobayashi, T. Matsuura, Y. Yano, T. Gake, K. Niitsu, N. Takai, T. J. Yamaguchi, "Low-Distortion Single-Tone and Two-Tone Sinewave Generation Using $\Sigma\Delta$ DAC", IEEE International Test Conference (poster session), Anaheim, CA (Sept. 2011)
- [7] 加藤啓介, 若林和行, 山田貴文, 小林春夫, 小林修, 新津葵一「任意波形発生器を用いた低歪み 2 トーン信号発生技術」第 24 回回路とシステムワークショップ. 淡路島 2011 年 8 月 2 日).
- [8] HV ブリッジレス PFC 開発用キット
<http://www.tij.co.jp/tool/jp/tmdshvblpfckit>
- [9] HV フェーズ・シフト・フル・ブリッジ開発用キット
<http://www.tij.co.jp/tool/jp/tmdshvpsfbkit>
- [10] TMS320C1x/C2x/C2xx/C5x アセンブリ言語ツール ユーザーズ・マニュアル, Texas Instruments (1996)

付録:PSFB DC/DC 周波数可変プログラム

CCS 開発環境で HVPSFB_VMC プロジェクトを開いて、下記のようにプログラムを追加と変更を行なった。

① HVPSFB_VMC の main.c ファイルに追加する function の申告

void PWMDRV_PSFB_PWM_CHANGE(int16 n,int m, int16 IoutR,int16 SR_Enable);

引数型説明

int16 n , PWM control register Pointer flag

int16 m, m=range,つまり電流範囲に指する flag

int16 IoutR ,出力電流 (今回の実験予め電流範囲を確定したので、今回不要ですが、以後のプログラム修正に必要可能性があるため、保留とします)

intSR_Enable ,synchronous rectification 許可 flag

②main.c の void C1(void)に上記の function がコールされる。

```
voidC1 (void)
//=====
=
{
if (auto_DB == 1)
{
switch (range)
{
case 0:
if (Gui_Iout< 640) // 5A
{
//////////Recode by GaoChuan
//////////

PWMDRV_PSFB_PWM_CHANGE(1,range,Gui_Iout,1);

//////////Recode by GaoChuan
//////////

/* if (dbAtoP_leg<36) dbAtoP_leg++;
```

```

else
{
    if (dbAtoP_leg>36) dbAtoP_leg--;
}

if (dbPtoA_leg<44) dbPtoA_leg++;
else
{
    if (dbPtoA_leg>44) dbPtoA_leg--;
}*/

dbAtoP_leg = 36;
dbPtoA_leg = 44;
range = 0;
}
else range = 1;
break;

case 1:

    if (Gui_Iout> 512 &&Gui_Iout< 1024) // 4A & 8A
    {

        ///////////////////////////////////Recode by GaoChuan
        ///////////////////////////////////

        PWMDRV_PSFB_PWM_CHANGE(1,range,Gui_Iout,1);

        ///////////////////////////////////Recode by GaoChuan
        ///////////////////////////////////

        /*
        if (dbAtoP_leg<32) dbAtoP_leg++;
        else
        {
            if (dbAtoP_leg>32) dbAtoP_leg--;
        }

```

```

        if (dbPtoA_leg<42) dbPtoA_leg++;
        else
        {
            if (dbPtoA_leg>42) dbPtoA_leg--;
        }*/

        dbAtoP_leg = 32;
        dbPtoA_leg = 42;
        range = 1;
    }
    else
    {
        if (Gui_Iout> 1024) range = 2;
        else range = 0;
    }
    break;

case 2:

    if (Gui_Iout> 896 &&Gui_Iout< 1408) // 7A & 11A
    {
        ///////////////////////////////////Recode by GaoChuan
        ///////////////////////////////////

        PWMDRV_PSFb_PWM_CHANGE(1,range,Gui_Iout,1);

        ///////////////////////////////////Recode by GaoChuan
        ///////////////////////////////////

        .....

    }

case 8:

    if (Gui_Iout> 3200 &&Gui_Iout< 4096) // 25A & 32A

```

```

{
    ///////////////////////////////////Recode by GaoChuan
    ///////////////////////////////////

    PWMDRV_PSFB_PWM_CHANGE(1,range,Gui_Iout,1);

    ///////////////////////////////////Recode by GaoChuan
    ///////////////////////////////////

/*
    if (dbAtoP_leg<16) dbAtoP_leg++;
    else
    {
        if (dbAtoP_leg>16) dbAtoP_leg--;
    }

    if (dbPtoA_leg<14) dbPtoA_leg++;
    else
    {
        if (dbPtoA_leg>14) dbPtoA_leg--;
    }*/

    dbAtoP_leg = 16;
    dbPtoA_leg = 14;
    range = 8;
}
else
{
    if (Gui_Iout> 4096) range = 9;
    else range = 0;
}
break;

case 9:

    if (Gui_Iout> 3968 &&Gui_Iout< 4864) // 31A & 38A
    {

```

```
//////////Recode by GaoChuan
//////////
```

```
PWMDRV_PSFB_PWM_CHANGE(1,range,Gui_Iout,1);
```

```
//////////Recode by GaoChuan
//////////
```

```
/*      if (dbAtoP_leg<15) dbAtoP_leg++;
      else
      {
          if (dbAtoP_leg>15) dbAtoP_leg--;
      }

      if (dbPtoA_leg<13) dbPtoA_leg++;
      else
      {
          if (dbPtoA_leg>13) dbPtoA_leg--;
      }*/

      dbAtoP_leg = 15;
      dbPtoA_leg = 13;
      range = 9;
  }
else
  {
      if (Gui_Iout> 4864) range = 10;
      else range = 0;
  }
break;

case 10:
    if (Gui_Iout> 4736 &&Gui_Iout< 5632)  // 37A & 44A
    {
        //////////Recode by GaoChuan
        //////////
```

PWMDRV_PSFB_PWM_CHANGE(1,range,Gui_Iout,1);

```

                ////////////////Recode by GaoChuan
                ///////////////////
/*
                if (dbAtoP_leg<14) dbAtoP_leg++;
                else
                {
                        if (dbAtoP_leg>14) dbAtoP_leg--;
                }

                if (dbPtoA_leg<12) dbPtoA_leg++;
                else
                {
                        if (dbPtoA_leg>12) dbPtoA_leg--;
                }*/

                dbAtoP_leg = 14;
                dbPtoA_leg = 12;
                range = 10;
        }
        else
        {
                if (Gui_Iout> 5632) range = 11;
                else range = 0;
        }
        break;

case 11:

        if (Gui_Iout> 5504)

        // 43A

        {

                ////////////////Recode by GaoChuan
                ///////////////////

```

PWMDRV_PSFB_PWM_CHANGE(1,range,Gui_Iout,1);

```

//Recode by GaoChuan
//
/*
    if (dbAtoP_leg<12) dbAtoP_leg++;
    else
    {
        if (dbAtoP_leg>12) dbAtoP_leg--;
    }

    if (dbPtoA_leg<12) dbPtoA_leg++;
    else
    {
        if (dbPtoA_leg>12) dbPtoA_leg--;
    }*/

    dbAtoP_leg = 12;
    dbPtoA_leg = 12;
    range = 11;
}
else range = 0;
break;
}

}
```

③ コールされる function の中身

```
/*
 * PWM_PSFB_PWM_CHANGE.c
 *
 * Created on: 2012/12/20
 * Author: sen.kou
 */
```



```

#include "PeripheralHeaderIncludes.h"

#include "DSP2802x_EPWM_defines.h"           // useful defines specific to EPWM

extern volatile struct EPWM_REGS *ePWM[];

void PWMDRV_PSFBS_PWM_CHANGE(int16n, int m, int16 IoutR, int16 SR_Enable)
{
    switch(m)
    {
        case 0:
            (*ePWM[n]).TBPRD = 858;    //70kHz
            (*ePWM[n]).CMPA.half.CMPA = 858/2;
            // Fix duty at 50%
            (*ePWM[n+1]).TBPRD = (858-1);
            (*ePWM[n+1]).CMPA.half.CMPA = 858/2;
            // Fix duty at 50%
            if (SR_Enable == 1)
            {
                (*ePWM[n+3]).TBPRD=858/2-1;

            }

            break;

        case 1:
            (*ePWM[n]).TBPRD = 750;    //80kHz
            (*ePWM[n]).CMPA.half.CMPA = 750/2;
            // Fix duty at 50%
            (*ePWM[n+1]).TBPRD = (750-1);
            (*ePWM[n+1]).CMPA.half.CMPA = 750/2;
            // Fix duty at 50%
            if (SR_Enable == 1)
            {
                (*ePWM[n+3]).TBPRD=750/2-1;
            }
    }
}

```

```

    }

case 2:

    (*ePWM[n]).TBPRD = 668;    //90kHz
    (*ePWM[n]).CMPA.half.CMPA = 668/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (668-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 668/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=668/2-1;

    }

    break;

case 3:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;

case 4:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

```

```

// Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;

case 5:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;

case 6:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;

```

```

case 7:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;

case 8:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;

case 9:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)

```

```

        {

            (*ePWM[n+3]).TBPRD=600/2-1;

        }

        break;
case 10:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;
case 11:

    (*ePWM[n]).TBPRD = 600;    //100kHz
    (*ePWM[n]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    (*ePWM[n+1]).TBPRD = (600-1);
    (*ePWM[n+1]).CMPA.half.CMPA = 600/2;

    // Fix duty at 50%
    if (SR_Enable == 1)
    {

        (*ePWM[n+3]).TBPRD=600/2-1;

    }

    break;
}

}

```